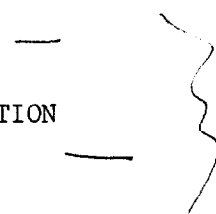RESEARCH PROGRAM:

BIOMEDICAL KNOWLEDGE REPRESENTATION

N LM.

Submitted to the
National Library of Medicine

February 1979

Department

ersity

Computer Science

Stanford Univ

TABLE OF CONTENTS

I.   THE PROGRAM PROPOSED

We begin this proposal with a description of the broad program contemplated, with rationale and justification of need, and a description of resources and facilities already available for the purpose.

Herein we propose a five-year program of research on knowledge representation, and the various problems associated with it in the design of knowledge-based computer programs. The Stanford University group will work collaboratively with a group from the University of Missouri's Health Care Technology Center, under the direction of Dr. Donald Lindberg. The program will be under the general direction of Professor Edward Feigenbaum of Stanford, who presently serves also as the Principal Investigator of SUMEX-AIM, the NIH-sponsored National Computer Resource for research on the application of Artificial Intelligence (AI) techniques to medicine and biology. This Resource will serve the computer needs of the proposed program.

The proposed program consists of four activities: three projects and a core research activity.

Projects One and Three address the problems of knowledge representation, acquisition, and utilization in specific medical/hospital settings.

In Project One, the clinical setting is the Oncology Day Care Clinic. The task that provides specificity and direction to the research is the construction of a consultation system regarding experimental protocols and selection of therapy for clinic outpatients. This project is led by Professor E.H. Shortliffe of the Stanford Medical School, the original developer of the MYCIN program for consultations regarding infectious disease diagnosis and therapy.

In Project Two, the transfer of such expertise to other places and to other medical applications can be viewed as the primary goal. One powerful way of cumulating the concepts and methods of an emerging branch of Computer Science is to cumulate them in working software packages that widely applicable and widely shared. This project aims at developing a number of such packages or "tools", constituting a computer-program "workbench" for further research on and application of knowledge-based systems. The packages emerge as generalizations of work done in the task-specific projects;constitute a very tangible type of result therefrom; and serve to amplify and accelerate future

1

efforts. This project is under the direction of Professors Bruce Buchanan and Douglas Lenat of Stanford.

In Project Three, the setting is the Clinical Laboratory and the task is one of acquiring and representing the medical expertise that allows the laboratory expert (e.g. the Laboratory Director) to interpret test results and discuss these with the patient's clinical physician. This is the inter-university collaboration headed by Dr. Lindberg. An important subgoal of this project is the transfer of the Stanford expertise in knowledge based systems research to the Missouri Center.

The Core Research Activity will' investigate a variety of fundamental research questions whose answers will shape present and future developments in knowledge representation research. Such questions involve formalisms and data structures for representing various types of knowledge; various methods—some automatic, some interactive—for acquiring new knowledge in systems; new inferential methods for putting this knowledge to work; strategy-knowledge representations for reasoning about the domain specific knowledge; and so on. The Core Research Activity is under the direction of Professor Feigenbaum. Douglas Lenat of Stanford.

Lastly, it is an objective of the overall program to disseminate the findings of the research, and to provide training opportunities to others. This objective will be accomplished through publications, presentations of research results at scientific meetings, by making room in the operational sites and the core activity for visiting scientists and trainees, and by - participation in a special annual meeting. The meeting to discuss our research and similar projects in this field will either be a part of or be coordinated with the annual artificial intelligence in medicine meetings at Rutgers University. That is, in years when the Rutgers meeting agenda and housing facilities can accommodate this group and its audience, we will join with Rutgers. In years when this is not possible, we will sponsor a separate meeting addressed to the four principal objectives of this program.

The administrative arrangements for the Program will be these:

The Principal Investigators of the various program activities will collectively constitute an Executive Committee for the Program, under the chairmanship of the Program Director. The Executive Committee will meet routinely by telephone-conference and occasionally face-to-face.

An Advisory Group will be formed, consisting of colleagues at other institutions who share our motivations and scientific interests. This group will advise the Executive Committee on major decisions and will offer peer review as necessary. The kernel of the Advisory Committee will be drawn from the membership of the SUMEX-AIM Advisory Committee (for which Dr. Lindberg is currently chairman).

I.A. Rationale for the Program

I.A.1. What do we mean by knowledge?

Computer scientists have long recognized that a computer is a general symbol-manipulating device. Arithmetic constitutes a special case of this capability—the manipulation of those symbols that are numbers. In this proposal we will be discussing non-numeric symbol manipulation by computers. In thinking about non-numeric computation, it is useful to think about:

a.   inference methods (as opposed to calculation and algorithms)

b.   qualitative "lines of reasoning" (as opposed to quantitative formulations)

c.   symbolic facts (not merely numeric parameters and formulas)

d.   decision rules of expertise and judgment (as opposed to mathematical decision rules)

The use of the term "knowledge" in this proposal is intended to cover both (c) and (d) above. In common usage, the term "knowledge" does not usually include (d), because such judgmental and experiential knowledge is largely tacit knowledge and therefore not recognized (i.e. the knowledge is "private" and the expert is not aware of what he/she knows and is using in problem-solving). The knowledge is private not because the expert is unwilling to share it, but because he/she is unable to discover and verbalize it.

It is central to our view that such knowledge—the knowledge of "expertise"— is critical for competent practice in medicine and science, in fact constituting the bulk of the knowledge employed in such practice. We view as a matter of great importance that such knowledge be codified and given a concrete (and at least semi-formal) representation, so that it can be used, stored, transmitted to others, analyzed, discussed, and taught.

Every activity of this proposed program is aimed at developing the scientific concepts and methods by which this can be most expeditiously, carefully, and usefully done.

Symbolic computation, though general and powerful, has hardly begun to be exploited in real applications. The specialty within Computer Science that has studied complex methods of symbolic computation is "Artificial Intelligence Research."

I.A.2.   Some Relevant Global and Local History

Early work in artificial intelligence aimed toward the creation of generalized problem solvers. Work on programs like GPS [by Newell and Simon] and theorem proving , for instance, was inspired by the apparent generality of human intelligence and motivated by the belief that it might prove possible to develop a single program applicable to all (or most) problems. While this early work demonstrated that there was a large body of useful general purpose techniques (such as problem decomposition into subgoals, and heuristic search in its many forms), these techniques did not by themselves offer sufficient power for expert levels of performance. Recent work has instead focused on the incorporation of large amounts of task specific knowledge is what have been called "knowledge-based" systems. Rather than non-specific problem solving power, knowledge based systems have emphasized high performance based on the accumulation of large amounts of knowledge about a single domain. A second successful focus in work on intelligent systems has been the emphasis on the utility of solving "real world" problems, rather than artificial problems fabricated in simplified domains. This is motivated by the belief that artificial problems may prove in the long run to be more a diversion than a foundation for further work, and by the belief that the field has developed sufficiently to provide techniques that can aid working scientists. While artificial problems may serve to isolate and illustrate selected aspects of a task, solutions developed for those selected aspects often do not generalize well to the complete problem.

There are numerous current examples of successful systems embodying both of these trends, systems which apply task-specific knowledge to real world problems.

The following are synopses of a variety of knowledge-based systems developed by the Stanford participants in this program over the past thirteen years:

DENDRAL: An intelligent assistant to an analytic and structural chemist. It infers the structures of complex organic molecules from structural constraints. These constraints are either supplied interactively by the user from his "private" knowledge and intuition, or are inferred automatically from instrument data, such as mass spectral data, nuclear magnetic resonance data, etc. For those families of molecules for which the knowledge base has been carefully elaborated, the DENDRAL program performs at levels equalling or exceeding the best human experts. The DENDRAL program now has a significant user community in university laboratories and in industry, and is being used to solve difficult real problems.

Meta-DENDRAL: This program is focused on the problem of elaborating DENDRAL's knowledge base for specific families of compounds. It infers an empirical theory (a body of fragmentation rules) of the mass spectrometry of specific families from recorded mass spectral data. It has not only "rediscovered" rules previously acquired from chemists, but has discovered novel rules for certain families—rules that have recently warranted publication in the chemical literature.

MYCIN: This program is an intelligent assistant to a physician diagnosing infectious diseases. In conjunction with its diagnoses, it recommends therapeutic action. It is capable of explaining its line-of-reasoning in any (and varying) level of detail to the user in English. It can accept new decision rules from the user in English. It keeps an updated model of its own knowledge base, which it uses to critique the introduction of new rules into the system. It is capable of acquiring and using measures of the uncertainty of the knowledge, and produces a "believability" index with each inference, i.e., it is capable of approximate implication. A version called EMYCIN, sans infectious disease knowledge, has been developed to extend the use of the system to other domains.

HASP: Project scientists working in a classified environment led the development of a signal-understanding program for continuous surveillance of certain objects of military interest. The program ran successfully in a number of highly

varied test situations, and is being further developed in a currently-funded ARPA program. The program used a design for incremental hypothesis formation that was a modification of the HEARSAY design for the CMU speech-understanding system. Symbolic knowledge from a number of sources was used to aid the interpretations of the primary signal data. Time-dependent analysis was novel in this system and played an important role.

AM: This remarkable program conjectures "interesting" mathematical concepts. Its knowledge base encompasses the (usually private) knowledge of a mathematician as to what constitutes an "interesting" construct in mathematics. Starting with the simplest set-theory concepts, and hundreds of rules defining "interestingness" of mathematical concepts, it has conjectured such concepts as addition, multiplication, factorization, primes, unique factorization into primes (the fundamental theorem of arithmetic), and an almost unstudied concept in number theory called "maximally divisible numbers."

MOLGEN: (under development) This program is being designed to be an intelligent assistant to an experimental molecular geneticist in formulating plans for laboratory experiments involving the manipulation of short DNA strands with restriction enzymes. The program is concerned with representing knowledge about planning and with the automatic formulation of plans to the level of detail demanded by the user. The program's knowledge must be represented at various levels—biological, genetic, topological, and chemical—and these levels must be incorporated into the reasoning.

CRYSALIS: Crystallographic Image Interpretation: (under development) This program is being designed to interpret ambiguous, incomplete three-dimensional image data obtained in x-ray crystallography of protein structures. The image input data is the so-called electron density map and the answer desired is an approximately correct protein molecule (or portion thereof). As with HASP, many sources of symbolic data support the interpretation of the primary signal data. The HASP program organization has been imported as a test of its generality. The interpretation problem is difficult because the best wavelength available (x-rays) is too long to resolve atoms and interatomic separations; hence the need for additional sources of symbolic knowledge, e.g., the amino acid sequence of the protein.

PUFF: This program interprets data from the pulmonary function testing laboratory and provides for the Lab Director an interpretive summary of findings regarding airways obstruction, lung restriction, and the degree of severity; subtype, such as bronchitis; the corroborating evidence and its weight; treatment

recommendations;etc.   This knowledge-based system was built in
collaboration with a pulmonary physiologist at Pacific Medical
Center, and is in routine daily use.

VM: A program that offers the attending physician or nurse
interpretations of streams of data monitored from a patient in
Intensive Care; signals alarm conditions due to unexpected
patient condition or possible instrument malfunction; and offers
advice regarding the management of the patient's ventilator
machine assistance. This is another collaboration with Pacific
Medical Center.

SACON: A MYCIN-like consultation system that advises a
structural engineer on the analysis plan necessary to compute the
multitude of structural engineering design parameters needed for
building a complex structure (such as an airplane wing or an off-
shore oil drilling platform or a building). Interactively, in
consultation, the user supplies the design specifications. The
system was built in collaboration with structural engineers at
the MARC Analysis Corporation. It was built rapidly using the
EMYCIN package discussed later.

In short, as the capsule sketches above indicate, the main
themes of our work involve: the acquisition and maintenance of
knowledge bases; the utilization of this knowledge in a variety
of ways for data interpretation, problem solving, and planning;
and the representation of this knowledge for computer inference.

I.A.3.   Knowledge Representation Issues and Designs--the
MYCIN Experience

In lieu of further general discussion of knowledge
representation, we have chosen to explicate in some depth our
viewpoint and methodology by drawing upon the experience in
design and development of just one of our programs, the well-
known consultation system MYCIN. For us, this work has been
seminal; hence the discussion of it that follows generalizes to
most of the other Stanford-based efforts mentioned above.

I.A.3.a.   Background

Several computer programs have been written that attempt to
model a physician's decision making processes. Some of these
have stressed the diagnostic process itself [27],[17]; others

7

have been designed principally for use as educational tools
[31],[36],[56]; while still others have emphasized the program's
role in providing medical consultations [4],[29],[51],[57].
Actually, these applications are inherently interrelated since
any program that is aimed at diagnosing disease has potential use
for educating and counselling those who lack the expertise or
statistical data that have been incorporated into the program.
Consultation programs often include diagnosis as a major
component, although their principal focus involves interactive
use by the physician and/or the determination of appropriate
advice regarding therapy selection.

In general, the educational programs designed for
instruction of medical students and other professionals have met
with more long-term success [60] than has been the case for the
diagnostic and consultation programs. The relative success in
implementing instructional programs may result because they deal
only with hypothetical patients as part of an effort to teach
diagnostic and therapeutic concepts, whereas the consultation
programs attempt to assist the physician in the management of
real patients in the clinical setting. A program making
decisions that can directly affect patient well-being must
fulfill certain responsibilities to the physician if he is to
accept the computer and make use of its knowledge.

Physicians will, in general, reject a computer program
designed for their use in decision making unless it is
accessible, easy to use, forgiving of noncrucial errors from
nonexpert typists, reliable, and fast enough to facilitate the
physician's task without significantly prolonging the time
required to accomplish it. They also require that the program
function as a tool to the physician, not as an all-knowing
machine that analyzes data and then states its inferences as
dogma without justifying them.

Those who design computer programs to give advice to
physicians must devise solutions to these requirements in an
effort to combat the current lack of acceptance of computer-aided
diagnosis by the medical profession [14],[24]. The physician is
most apt to need advice from such a program when an unusual
diagnostic or therapeutic problem has arisen. However, he may be
unwilling to experiment with a program that does not meet the
general requirements outlined above.

Considerations such as those mentioned here have in large
part motivated the research of our group over the last half-
decade. We felt it was important to devise a consultation
program that was (1) useful, (2) educational when appropriate,
(3) able to explain its advice, (4) able to understand and

respond to simple questions stated in natural language, (5) able to acquire new knowledge interactively, and (6) able to be modified easily. Although we recognized that this list of design considerations was somewhat idealistic in light of the state of the art in computer science, we did feel that it provided a useful set of long-range goals. The program we developed, known as MYCIN, has had considerable success in achieving many of the goals stated.. The current research proposes to build on the MYCIN experience, both by expanding the basic computer science methodology to deal with recognized problems as yet unsolved, and by implementing a consultation system in a clinical setting where its usefulness and acceptability to physicians can be assessed.

I.A.3.b.   The MYCIN Program

As medical knowledge has expanded in recent decades, it has become evident that the individual practitioner can no longer hope to acquire enough expertise to manage adequately the full range of clinical problems that will be encountered in his practice. Thus when a patient's problem clearly falls outside the area of the attending physician's expertise, consultations from experts in other subspecialties have become a well accepted part of medical practice. Such consultations are acceptable to doctors in part because they maintain the primary physician's role as ultimate decision maker. The consultation generally involves a dialog between the two physicians, with the expert explaining the basis for his advice and the nonexpert seeking justification of points he finds puzzling or questionable. A consultant who offered dogmatic advice he was unwilling to discuss or defend would find his opinions were seldom sought.

Fig. 1 shows a schematic view of the consultation process. Appendix A shows a detailed typescript of a sample consultation. The physician nonexpert gives information about his patient to the expert in response to questions and, in return, receives advice and explanations. Thus there are actually three kinds of information flow between the physician and his consultant. The MYCIN program models the consultative process by attending to all three kinds of information. It is our conviction that programs which ignore the explanation pathway will fail to be accepted by physicians because they will see in such systems too severe a departure from the human consultation process (in which the primary physician is provided with sufficient information to allow him to decide whether to follow the offered advice).

9
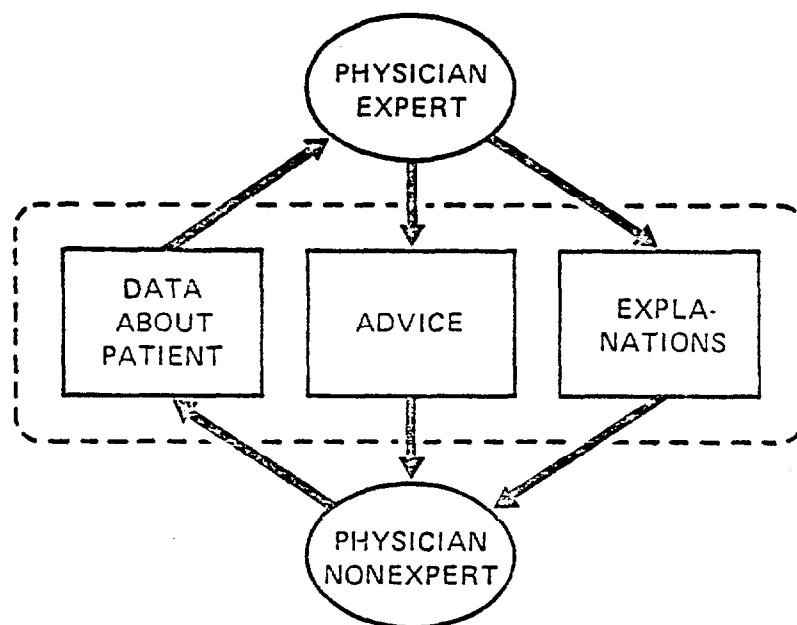
Figure 1 - Information Flow Between Physician And Consultant

MYCIN is a LISP program designed to serve as a clinical consultant on the subject of therapy selection for patients with serious infections. The program may be envisioned as interposed between the expert and nonexpert in much the way that the large box is positioned in Fig. 1. The difference is that the human expert can offer only general knowledge to the program, not patient-specific decisions. The program thus becomes the decision maker, using general medical knowledge from experts to assess a specific patient and to give advice plus explanations for its judgments.

Fig. 2 details the organization of MYCIN relative to the human consultation process depicted in Fig. 1. As before, the nonexpert offers data about his patient and in return receives both advice and, when desired, information via one of two internal explanation mechanisms (the general question-answerer or the reasoning-status checker). The basis for all decisions is domain-specific knowledge acquired from experts (static knowledge). A group of computer programs (the rule interpreter)

uses this knowledge, and data about the specific patient, to
generate conclusions and, in turn, therapeutic advice. It
simultaneously keeps a record of what has happened, and this
record is available to the explanation routines if the physician
asks for justification or clarification of some conclusion that
the program has reached. Although Fig. 2 is somewhat
complicated, the following discussion should clarify the
interrelationships among the various system components depicted
in the diagram. Furthermore, Appendix A gives detailed examples
of all the features described below.

Knowledge Representation

Static Knowledge

Static knowledge refers to all data that are constant in
the program and unchanging from one consultation to the next.

Facts About The Domain. Much of the knowledge MYCIN
requires is simple statements of fact about the domain. These
can generally be represented as attribute-object-value triples
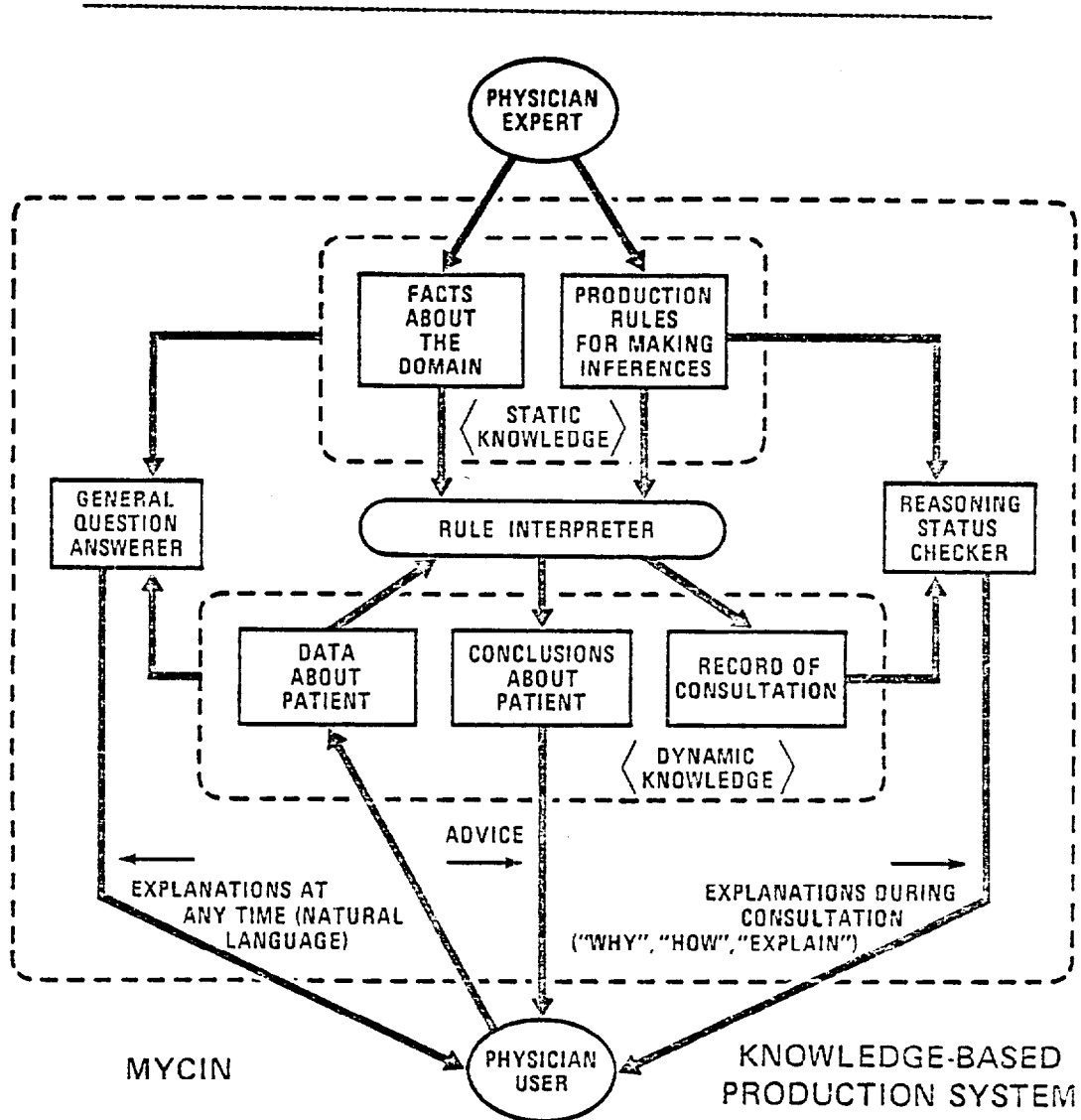
Figure 2 - Schematic Description Of MYCIN Related To Fig. 1

Production Rules. (Appendix A - Section I) In addition to simple facts, MYCIN requires judgmental knowledge acquired from experts and available for use in analyzing a new patient. Judgmental knowledge in MYCIN is expressed as production rules [16] which define certain preconditions (the PREMISE) that allow a conclusion to be reached (the ACTION) with a specified degree

of confidence (the "certainty factor" [49]). Although such rules are stored as LISP list structures, a series of routines is available for translating them into English. For example:

> PREMISE: If the stain of the organism is gramneg, and
> the morphology of the organism is rod, and
> the organism is anaerobic,
> ACTION: Then there is suggestive evidence (.7) that
> the identity of the organism is bacteroides.

Note that the purpose of this rule is the determination of organism identity. Rules are classified and accessed in accordance with their purpose as described below.

Dynamic Knowledge

Dynamic knowledge refers to all data that are variable and change from one run of the program to the next.

Data About The Patient - Acquired From The User. MYCIN asks questions of the user, driven by a reasoning algorithm described below. These questions generally ask the user to fill in the "value" in an attribute-object-value triple (eg., "What is the patient's name?"), or to give the truth value of a predicate (eg., "Is the patient a compromised host?"). Thus these data may be represented, once acquired, in precisely the way that facts about the domain are represented in the static knowledge base (see above).

Data About The Patient - Generated By The Program. When the preconditions in the PREMISE of a rule are found to hold, MYCIN executes the ACTION portion of the rule and generates a new "fact" which can, once again, be represented as an attribute-object-value triple. As mentioned above, conclusions may also have a confidence value associated with them, thereby requiring that the triple be expanded to a quadruple:

the identity of ORGANISM-1 is
        bacteroides, with   -
        certainty factor of 0.7
                (IDENTITY ORGANISM-1 BACTEROIDES .7)

Predicates may be similarly expanded. Furthermore, by generalizing this scheme to include representation of data acquired from the user, the physician may be asked to express his confidence in the answer he gives when MYCIN asks a question.

Maintenance Of A Record Of The Consultation. A history of the consultation is the third variety of dynamic knowledge. The details of representation need not be described here, but these data include records of which rules succeeded, which rules were tried but failed, how specific decisions were made, how information was used, and why questions were asked.

The Production System

The Rule Interpreter

This series of routines analyzes rules in the static knowledge base, determines whether they apply to the patient under consideration, and if so draws the conclusions delineated in the ACTION portions of the rules. This process would quickly become unmanageable as system knowledge grew if there were not a mechanism for selecting only the most relevant rules for a given patient. This is accomplished by a goal-oriented approach that we have described in detail [50],[51]. Briefly, as the rule interpreter examines the PREMISE of a rule, it notes whether the relevant data needed to determine the truth of each precondition are already known. If not, it digresses to examine those rules which make conclusions about the data that are needed by the first rule. The PREMISE conditions of those rules may, in turn, invoke additional rules, and in this way a reasoning network relevant to the first rule is formed. Since rules are classified according to their purpose, as previously described, it is easy to identify all rules that may aid in determining the truth of a specific precondition. The entire process is initiated by invoking a specific "Goal Rule" which defines MYCIN's task and is the only rule necessarily invoked for every consultation. When MYCIN can find no rules for determining the truth of a precondition, it asks the user for the relevant data. If the physician does not know the information either, the invoking rule is simply ignored.

Maintenance Of Initiative In The Hands Of The Physician

As was discussed above, a physician is not likely to accept a system such as MYCIN if the program simply asks a series of

14

questions and then presents a piece of dogmatic advice as it terminates execution. The production system has therefore been provided with a series of "interrupts" that allow the physician to digress with questions of his own or to demand justification for the line of questioning on which MYCIN has embarked during the consultation. Whenever the program asks a question, the user can temporarily refuse to answer and instead call on the explanation capabilities described in the next section.

Explanations

The Reasoning-Status Checker (RSC)   (Appendix A - Section IV)

This component of the explanation system deals with most questions that arise during the consultation session itself. Because the context of current reasoning about the patient is well-defined, the physician can be given a great deal of information on the basis of a few simple commands that do not require natural language processing. These commands are briefly described below; the details of their implementation have also been documented [48]. As shown in Fig. 2, the reasoning status checker (RSC) uses only the knowledge base of rules and the current record of the consultation; the general question-answerer (GQA) described below, on the other hand, has access to all static and dynamic knowledge.

The WHY Command. Whenever MYCIN asks a question, the physician may prefer not to answer initially and instead to inquire about the reasoning underlying the questioning. Thus he may simply respond with the command WHY (i.e., "Why do you think that the information you are requesting may be useful?"). Since all questions MYCIN asks are generated by rules, and since the rules are selected according to their purpose as previously mentioned, an English language translation of the rule under consideration generally serves as an adequate response to the WHY query. The RSC therefore responds by displaying the current rule. In addition, it places an identifying number before each of the preconditions in the PREMISE and indicates whether the condition is (a) already known to be true, or (b) still under investigation (note that one of the latter group of preconditions will have generated MYCIN's current question to the user). The physician can in turn inquire why the displayed rule was selected by asking WHY a second time, and the RSC will accordingly display the next rule in the reasoning network.

The HOW Command. As mentioned above, when MYCIN displays a

rule in response to the WHY command, it labels each precondition in the PREMISE with a unique number. The physician may then respond to the displayed explanation by entering HOW followed by one of the identifying labels. If the reference condition is one that MYCIN has already concluded to be true, the RSC assumes that the physician is asking "HOW did you decide that the specified precondition is true?" and answers by citing the relevant rules used to make the decision. If, on the other hand, the cited condition has not yet been fully investigated, MYCIN assumes the physician is asking "HOW will you decide if the specified precondition is true?" and responds by citing the rules it intends to try, only some of which may actually succeed.

The General Question-Answerer (GQA) (Appendix A – Section V)

The general question-answerer (GQA) is a more comprehensive explanation system which, at any time during or after the consultation session, has full access to all static and dynamic knowledge in MYCIN (Fig. 2). Since it cannot make simple assumptions based on context, as the RSC can do, the GQA must accept and answer questions expressed in natural language. MYCIN's rule-based knowledge representation scheme, and some techniques borrowed from early work in computational linguistics [13],[30],[47], permit a straightforward but powerful approach to interpreting simple English questions without contending with several of the complex problems of natural language understanding. The details of this approach have been documented [76].

Questions About Static Knowledge. The ability to retrieve information from the static knowledge base gives the GQA a tutorial capability. Since the static knowledge is acquired from experts, the GQA can essentially act as an intermediary between an expert and a physician seeking general information about the infectious disease field. The user might ask simple questions of fact (eg., "Which culture sites are normally sterile?") or questions regarding judgments stored in rules. Questions of the second variety are termed "rule-retrieval" questions because they may be answered simply by identifying and displaying English versions of relevant rules from the knowledge base. Retrieval may be keyed to the rule PREMISE (eg., "How do you use the gram stain of an organism?"), the ACTION (eg., "When do you decide an organism might be a streptococcus?"), or to both the PREMISE and ACTION (eg., "Do you ever use the morphology of an organism to determine its identity?"). Furthermore, a question may deal with a specific rule (eg., "What is rule037?"). Note that none of these rules refers to a specific patient or consultation and thus requires no access to the dynamic knowledge base (Fig. 2).

Questions About Dynamic Knowledge. Although the RSC permits inquiries regarding the dynamic knowledge base, its scope is limited by the context of the current question being asked by MYCIN. If the physician wishes to ask more general questions regarding the status of MYCIN's reasoning, or if he wishes to review the program's decisions after the consultation is complete and MYCIN is no longer questioning him, the GQA gives him free access to all information about the specific consultation. Once again, the user might ask simple questions of fact (eg., "From what site was culture-2 obtained?") or questions regarding the basis for MYCIN's judgments. The second variety is again a rule-retrieval question, but is keyed to the consultation record in dynamic data rather than to the knowledge base of rules in static data (see Fig. 2). Thus questions may again reference the PREMISE (eg. "How did you use the gram stain of organism-1?"), the ACTION (eg., "What makes you think that organism-2 might be a streptococcus?"), or both (eg., "Did you use the morphology of organism-1 to determine its identity?"). Note that these questions parallel the examples given in the previous section but that they are consultation-specific and thus request the retrieval not of all relevant rules, but only those that were actually used successfully in the specified context. Finally, one may again wish to ask about a specific rule (eg., "Did you use rule037 when considering organism-1?").

Knowledge Acquisition

The only component of Fig. 2 not yet discussed is the crucial step of acquiring domain-specific knowledge from experts and coding it for storage in the static knowledge base. When MYCIN was first being developed, such knowledge was acquired by extensive meetings during which infectious disease experts and computer scientists discussed specific patients and attempted to analyze and extract the individual facts and rules that they were using. Recently extensive work has been devoted to the problem of automating the knowledge acquisition process in sessions involving clinical experts interacting with MYCIN directly (Appendix A - Section IX). This problem has been the subject of a doctoral dissertation by one member of our group [15].

Certainty Factors

Efforts to develop techniques for modeling clinical decision making have had a dual motivation. Their potential clinical significance has of course been apparent. The design of such programs also has required an analytical approach to medical reasoning that has in turn led to a distillation of decision criteria that in some cases had never been explicitly stated before. It is a fascinating and educational process for experts

to reflect on the reasoning steps that they have always used when providing clinical consultations.

Several programs have successfully modeled the diagnostic process [27],[28],[55]. Each of these examples has relied upon statistical decision theory as reflected in the use of Bayes' Theorem for manipulation of conditional probabilities. Use of the theorem, however, requires either large amounts of valid background data or numerous approximations and assumptions. The successful performance of Gorry and Barnett's early program [27], for example, and a similar study by Warner using the same data [55], depended to a large extent upon the availability of good data regarding several individuals with congenital heart disease. Gorry [28] has had similar access to data relating the symptoms and signs of acute renal failure to the various potential etiologies.

Although conditional probability provides useful results in areas of medical decision making such as those mentioned, vast portions of medical experience suffer from so little data and so much imperfect knowledge that a rigorous probabilistic analysis, the ideal standard by which to judge the rationality of a physician's decisions, is not possible. It is nevertheless instructive to examine models for the less formal aspects of decision making. Physicians seem to use an ill-defined mechanism for reaching decisions despite a lack of formal knowledge regarding the interrelationships of all the variables that they are considering. This mechanism is often adequate, in well-trained or experienced individuals, to lead to sound conclusions on the basis of a limited set of observations.

We have examined the nature of such nonprobabilistic and unformalized reasoning processes, have considered its relationship to formal probability theory, and have proposed a model whereby the incomplete "artistic" side of the practice of medicine might be quantified. We have had to develop this model of inexact reasoning in response to MYCIN's needs; i.e., the goal has been to permit the opinion of experts to become more generally available to nonexperts. The model is, in effect, an approximation to conditional probability. Although conceived with MYCIN's problem area in mind, it is potentially applicable to any domain in which real world knowledge must be combined with expertise before an informed opinion can be generated. The model has been described in detail [75] and is based upon a scheme of weighted numbers we call "certainty factors". Although the model has been implemented in the MYCIN system, and in EMYCIN (see below), and although it has allowed the program to demonstrate impressive decision making performance, we still recognize many problems with the formalism. The model has generated considerable attention in the literature [1] and many important suggestions for further research have been forthcoming.

18

Evaluations Of MYCIN's Performance

Work on MYCIN to date has concentrated on the infectious disease subfields of bacteremia and meningitis. Formal evaluations have been undertaken which show that MYCIN compares favorably with infectious disease experts in selecting therapy for patients with bacteremia [62] or meningitis [63]. However, we have not undertaken a clinical implementation of MYCIN yet, and do not intend to do so in the near future. The reasons for this decision are important in that they explain part of the reason that we have turned from infectious diseases to oncology at this time.

First, we have felt it is crucial that MYCIN not be placed on the wards for clinical use if it does not already compare favorably with other forms of consultative advice available to primary care physicians. We have learned that this requires that MYCIN know about essentially all major infectious disease subfields since the various disease syndromes interrelate clinically in such important ways. In our evaluations of the program, it has tended to be in those cases in which a concomitant infection existed at some other site that MYCIN has failed to perform adequately. Yet the time required for us to develop the required knowledge bases for genitourinary infections, endocarditis, pneumonia, and pelvic infections would necessarily be at least as long as the period it has required to acquire and test the system's knowledge of bacteremia and meningitis. We therefore anticipate a considerable period of time before the program will be able to provide consistently reliable infectious disease consultations and hence be ready for ward implementation.

There are other problems as well that have been brought out by the complex decisions involved in infectious disease therapy selection. First, the truth model we have devised (see discussion of certainty factors above) has several recognized inadequacies that will require further research and testing. Secondly, no computer-based decision making program with which we are familiar has adequately managed time relationships amongst variables, and MYCIN is no exception. We see the need for continued research into the ways in which the production rule formalism can be suitably adapted to accommodate the need to represent time dependencies in clinical reasoning and to use such dependencies to make appropriate decisions. For example, trends in a fever or white count over time may be much more important in assessing an infected patient's illness than the actual values of these parameters at the precise time when the consultation is being requested.

Finally, in order to expand MYCIN's infectious disease
knowledge into new problem areas, improved capabilities for
knowledge acquisition would be extremely useful. Although we
have made important initial steps in the development of this kind
of complex capability [15], there is clearly much more to be done
before an infectious disease expert who is a computer novice will
be able to comfortably interact at a computer terminal in order
to "teach" MYCIN the infectious disease judgmental knowledge that
it needs to know.

I.B.   Resources that exist to aid this project

The research work proposed herein will not stand alone or
apart from other research already under way in the two sites. The
personnel and facilities in place at the University of Missouri's
Health Care Technology Center are described later in the
appropriate Project section. At Stanford there is an interlocking
set of existing grants and contracts supporting the work of a
large group of scientists and students, the Heuristic Programming
Project of the Stanford Computer Science Department. This group
has, over the years, produced the various systems summarized
earlier.

Historically the most significant sources of funding have
been:

1. contracts from the Defense Advanced Research Projects
Agency, the leading government agency for funding artificial
intelligence research.

2. grants from the Biotechnology Resources Program of NIH
for the SUMEX-AIM computer facility, without which it would have
been very difficult to accomplish what was accomplished.

The other grants have had a short-term character. Some have
been renewed, others not.

The proposed NLM grant is important to this complex of
funding not only because it represents a significant amount of
funding but most importantly because it represents stable funding
over a five year period. It, therefore, like the ARPA funding,
will constitute the stable base of support that will allow the
work to advance steadily without personnel and funding
fluctuations. The NLM-sponsored work will, in turn, benefit from

the other supported work in the usual coordinated and synergistic way that significantly amplifies the effect of the NLM support.

The grant for the SUMEX-AIM computer resource ends in mid-1981. There is no reason now to believe that at renewal time the grant will face trouble. However such large facilities grants are always subject to a great deal of pressure, not always from peer-review. The need to service the research activities of an ongoing five year NLM research project will definitely add strength to the renewal application.

Finally, a resource of the greatest significance for the success of this work are the collaborative links that we have built over the years with medical scientists and clinicians at the Stanford Medical Center, the Pacific Medical Center, and the University of Missouri. It takes years to make such links work smoothly, but the resource is indispensable to a project on biomedical knowledge representation.

I.C.   Significance

Collectively, we stand on the threshold of a new era in our understanding of the nature of medical and scientific knowledge, its distribution, and its effective use. Superficially, the cause of this has been the emergence of electronic symbol-processing and digital communication. More substantially, the reason for optimism is the emergence of knowledge-based computer systems research and application as a viable scientific and technical discipline.

We are now beginning to understand in a scientific and technical way what practitioners have always understood about their fields of learning and practice: that the bulk of the knowledge they employ is not the knowledge of textbooks and journals, but the informal and judgmental knowledge gained from long experience and practice. This knowledge is almost never codified, but is passed from mentor to apprentice by long periods of training and interaction, such as the internship, residency, and the Ph.D. graduate program

In the last decade there have been significant demonstrations that such heuristic knowledge can be explicated, represented, and put to use. Needed is an interdisciplinary team consisting of computer scientists, domain specialists, and various computer programs and computer-oriented methodology.

Once explicated, this knowledge can participate in the ordinary processes of cumulation of understanding in a field. For example, it can be subject to further analysis and be the basis for empirical studies and experimental investigation. It can be criticized by peer review. And it can be taught, or disseminated by library methods (electronic or otherwise).

In addition, the formal knowledge of a field can be coupled to the informal knowledge to produce computer programs that act as "intelligent agents" to assist practitioners in solving large numbers of routine problems, and even some of the more difficult problems, with which they are faced. Some methods of computer-based inference are available today to do this, and more are coming as research in this area matures. The concept is one of "active knowledge" available to work for users, in contrast to the passive knowledge of texts and articles (knowledge which is useless until "discovered" by the practitioner through library search and reading).

Such a prospect is not visionary. It demands our immediate attention. We have known for many decades that computers are general symbol processing devices, not merely calculators. We have known for two decades how to program them to infer lines-of-reasoning through complex problems of a symbolic nature. In the last decade we have learned how to make such reasoning powerful and useful—by supplying such programs with considerable bodies of knowledge about the problem domains. And we have had to learn how to represent the knowledge. Now microelectronics has brought the time of low-cost computing upon us. The electronic processing necessary to make the power of symbolic computing available to a wide community will be available. We should not allow ourselves to drop behind in the development of the concepts and methods necessary for the emergence of the applications.

There are also roles for knowledge-based symbolic computing that are visionary, but must be explored. The kind of "active" knowledge we have been discussing can be used to assist in the discovery of new knowledge. The very human process of discovery of new knowledge is a slow and halting process at best, done by very few and marked by very rare bursts of creative insight. It now seems possible (even plausible) that models of certain kinds of discovery can be formulated that will systematize for computer application the intertwined activities of inferential search and literature (i.e. knowledge) search. The Meta-DENDRAL program (that has formulated new rules of fragmentation in mass spectrometry) and the AM program (that conjectured some not-so-new objects and theorems in number theory) are demonstrable precursors of this type of knowledge-acquiring program.

22

We envision a National Library of Medicine that will be a living library of the knowledge of medicine and biology, not merely the repository of texts, journals, and articles and not merely the immense file of their electronic images available at terminals.

II.   CODIFICATION   AND   USE   OF   MEDICAL   KNOWLEDGE   IN
ONCOLOGY

II.A.   Introduction

II.A.1.   Objectives

The long term objective of our research effort is the
development of tools for the representation and use of medical
knowledge in computer-based clinical consultation systems. Such
systems will provide useful assistance to primary care physicians
while incorporating features that heighten the acceptability of
the systems to their intended users. We also wish to increase
our understanding of the logic of medical diagnosis and therapy
planning through this work. To that end we propose a five year
research effort with the following goals:

(1) to demonstrate that a rule-based consultation system
with explanation capabilities can be usefully applied and gain
acceptance in a busy clinical environment;

(2) to improve the tools currently available, and to
develop new tools, for building knowledge-based expert systems
for medical consultation;

(3) to establish both an effective relationship with a
specific group of physicians, and a scientific foundation, that
will together facilitate future research and implementation of
computer-based tools for clinical decision making.

The basic research will build on our group's prior
experience with a computer-based consultant, termed MYCIN, that
uses production rule symbolic reasoning techniques to assist in
therapy selection for patients with serious infections. The
domain we have selected for the first clinical implementation of
these techniques is the management of research therapy protocols
for cancer outpatients at Stanford Medical Center's new oncology
day-care center.

II.A.2.   Background

This research builds on a long history of work on the MYCIN and EMYCIN projects directed principally by Shortliffe and Buchanan. Many of the persons developing those systems will be involved with the research proposed here. These two projects are described elsewhere and thus need not be described here as well.

II.A.2.a.   Stanford Division Of Oncology

In the past decade chemotherapy has assumed a more important role in the treatment of patients with cancer. Some 2,000 patients are under the direct care of the five faculty physicians of Stanford's Division of Oncology in the Department of Medicine. Most patients are receiving care on an outpatient basis, either at the Debbie Probst Oncology Day Care Center in Stanford Hospital or at the Division's twice-weekly clinic at the Palo Alto Veterans Administration Hospital. Altogether, about 9,000 outpatient visits are made to the Division physicians each year.

Effective management of cancer often involves more than one therapeutic technique. Increasingly, the initial course of treatment utilizes a combined modality approach. Surgery and/or radiation may be followed by chemotherapy to control any remaining cancer. However, chemotherapy alone may be curative in some cases.

Refined programs (protocols) have been developed for the administration of radiation and chemotherapy for many forms of cancer. The Division has had particular success with those used against Hodgkin's disease (the sixth most common cancer) and other lymphomas. In designing and carrying out individual programs of treatment, the physicians of the Division of Oncology work closely with Stanford specialists in other areas, particularly radiotherapists, surgeons, pathologists, diagnostic radiologists, pharmacologists, and immunologists. Stanford's expertise in these many disciplines contributes to the high level of care received by patients in the Division of Oncology.

The Division is of course also involved in educating and training physicians on all levels, from medical students to practicing physicians. Among the trainees are nine clinical fellows in oncology who participate actively in both clinical research and patient care. Five physician specialists and private physicians are involved directly with patient care in the

Debbie Probst Day Care Center.   Numerous others   participate in
the protocol studies.

The Division of Oncology also firmly believes that
excellence in patient care and in teaching programs is best
achieved where there is a continuing pursuit of new knowledge.
Each of the six full-time faculty members in the Division is
actively engaged in cancer research. The clinical research
efforts are concerned with the refinement and development of more
effective methods of treatment. New chemotherapy is being sought
and tested. Better combinations of chemotherapy, and of
chemotherapy with other methods (surgery, radiation,
immunotherapy), are also being developed.

Debbie Probst Oncology Day Care Center

The Division's new, modern, outpatient clinic was designed
in response to the physical and emotional needs of cancer
patients undergoing chemotherapy. Located on the lower level of
the Stanford Hospital, it is designed as a self-contained unit,
convenient and comfortable for both patients and attending
medical personnel.

Three kinds of treatment rooms are provided, including some
for observation or for lengthy (six to eight hours) infusions
that formerly had required hospitalization. Efficient service to
patients is facilitated by a television monitoring system (see
discussion of Motorola system below), a computer-based medical
record system (see discussion of TOD below), and facilities for
preparing chemotherapy, analyzing blood , and viewing x-rays.

Information Display System

When the Oncology Day Care Center was designed, plans were
made for an automated scheduling and information display system.
This system was developed in conjunction with the Motorola
company and is now in operation in the clinic. The
microprocessor-based system signals alphanumeric video
information to remote locations via video cables. Scheduling
secretaries keep appointment records on an associated floppy
disc, and on any given day four video display monitors in the
oncology conference room are used to display the day's schedule,
relevant lab test results for the outpatients being seen that
day, room assignments, and the name of the oncologist who will be
attending each patient. At present all data are entered by
secretarial personnel and there is no hands-on interaction
between the physicians and the small computer.

Time-Oriented Databank (TOD)

For the last several years the Division of Oncology has also been using the time-oriented record keeping system (TOD) originally designed by Dr. J. Fries for use in the Stanford Immunology Clinic [25],[58]. The data and all TOD programs are stored in the Stanford campus computer facility, an IBM 370/168. The emphasis in the design of the TOD system has been the analysis of large amounts of data on a body of similar patients, not on interactive record keeping in the clinical setting itself. Thus there are large amounts of data on Stanford oncology patients, stored by dates of clinic visits, kept on this distant computer for retrospective analysis. TOD provides several programs for statistical analysis of correlations, assessing prognosis by attribute matching, and assisting with other tasks that have traditionally required arduous chart review. Since the data are not currently being used for the care of individual patients, there may be a time lag of weeks before transcriptionists extract the relevant data from paper-based oncology outpatient charts and enter them into the TOD databank.

Oncology Treatment Protocols

As mentioned above, the Division of Oncology is active in clinical research and has many patients being treated under research protocols. There are currently about 30 operational protocols, about half of which are active in the sense that several patients are enrolled in the treatment plan at any given time. Many of the protocols are designed and overseen by Stanford oncologists, but there are also cooperative studies involving Stanford and several other institutions. In many cases, the cooperative studies are overseen by the Northern California Oncology Group (NCOG) which has its headquarters very near the Stanford campus. Each protocol is described by a lengthy article, often 45-60 pages, that explains the justification for the therapeutic approach, outlines criteria for patient selection for the study, describes therapeutic options, and details the specific chemotherapy doses, dose modification, and laboratory and clinical data that must be obtained on each visit. It is quite impossible for any single individual to know the details of all 30 protocols. This is a particularly great problem because the physicians seeing oncology outpatients include fellows, residents, and medical students; these individuals have limited oncology experience and, in the case of house staff and students, generally rotate through oncology for only 4-8 weeks at a time. (See [41] for discussion of one approach which emphasizes use by primary care physicians, but has not emphasized a well-designed human interface.)

II.A.3.   Rationale

    The rationale for the proposed research has largely been described in previous sections. In short, there has been limited success of statistical, data retrieval, and decision analysis programs in dealing with the judgmental knowledge of expert physicians and the uncertainties of medical data. We have made encouraging strides in the development of symbolic reasoning techniques for application to clinical decision making and believe that the time is now appropriate for the clinical implementation of such a system. Only then will it be possible to assess the power of capabilities which have been designed to make consultation systems acceptable to physicians. Although we recognize that the short term impact of such systems is limited by the current state of the art in computer science, the impetus for appropriate basic research and development of new interactive techniques will come largely through the lessons learned in undertaking clinical implementations. Since techniques already exist that have potential for considerable short-term clinical impact, we believe it is now appropriate to spend part of our time on a project for clinical use.

    Although our interest is in the development of systems for offering any kind of subspecialty expertise to primary care practitioners, the initial application selected has been the management of complex therapy protocol information in an outpatient oncology clinic. This domain was selected for a number of reasons:

    (1) There are large amounts of information in the protocols but relatively little inferential complexity; those problems that have prevented us from attempting clincal implementation of the MYCIN System for infectious diseases can therefore largely be avoided.

    (2) There is a small core of faculty members and oncology fellows who are largely responsible for patients in the day care center. Hence a relatively small number of individuals will need to be introduced to the consultation system, and their continuing roles in the clinic will heighten their chances of becoming comfortable with computer-based techniques.

    (3) There is already an awareness of, and involvement with, computers in the Oncology Day Care Center (in the form of the information display system previously described and associated video display monitors). Thus, although there is not yet hands-on computer use by oncologists in the clinic, computer-related hardware is evident and accepted by the clinicians at the outset

of the proposed research. Many fellows and faculty also use the
TOD system for clinical research and thus have limited, but very
positive, experience with computer use.

      (4) Although the application of symbolic reasoning
techniques to the protocol management problem will not tax many
of the capabilities we have developed in the MYCIN context, it is
precisely this simplicity which makes the problem appealing as a
first clinical venture. If the information handling task can be
implemented relatively easily within the EMYCIN formalism, as we
expect it can, then we will be able to concentrate initially on
issues of making the system's reasoning and knowledge base
understandable as well as making the system's interaction
acceptable to physicians.

      (5) The initial investment in establishing a role for
interactive computing in the oncology outpatient setting at
Stanford will have considerable potential for facilitating
interactions between our protocol management system and the
Division of Oncology's current computer-related efforts (the
information display system, and the time-oriented databank). We
envision some challenging extensions to the consultation program
whereby physicians interacting with the protocol management
system may simultaneously benefit from direct connections between
our computer and the other oncology systems.

II.B.   Specific Aims

We propose core research as well as new demonstrations of
the clinical usefulness of present capabilities developed under
MYCIN research.

As has been discussed, we have identified an important
clinical problem in the outpatient oncology clinic at Stanford,
and have begun a collaboration with members of the oncology
division to develop and implement a Protocol Management System
(PMS) for use in the oncology clinic.   Our proposal is to
demonstrate   that computer-based   reasoning   and interactive
techniques developed during MYCIN research can be effectively
applied to an important clinical problem, namely the management
of oncology protocol data.

The infectious disease domain with which we have been
involved to date involves complex reasoning   and computing
problems that we feel prevent the short term development of a
clinically useful infectious disease consultation system. The
oncology problem, on the other hand, involves large amounts of
knowledge but rather simple reasoning that current techniques
should be able to manage effectively.   The complexities of
infectious diseases, however, have provided   a particularly
appropriate domain for devising new computing approaches while
analyzing clinical reasoning.   These difficult problems remain
major research interests of our group. We propose spending
approximately half our time continuing to work on basic tools for
expert medical consultation systems, using the current content of
the infectious disease knowledge base without any efforts to
extend its scope in the short term.

Specifically, our aims during the five years of proposed
research are:

Artificial Intelligence Objectives

(1) To implement and evaluate recently developed techniques
designed to make computer technology more natural and acceptable
to physicians;

(2) To extend the methods of rule-based consultation
systems to interact with a large database of clinical
information;

(3) To continue basic research into the following problem areas: mechanisms for handling time relationships, techniques for quantifying uncertainty and interfacing such measures with a production rule methodology, approaches to acquiring knowledge interactively from clinical experts. These are some of the problems we have identified that have prevented the MYCIN infectious disease application from being clinically implemented as yet.

Oncology Clinic Objectives

We plan to develop and implement a Protocol Management System (PMS), for use in the oncology day care center, with the following capabilities:

(1) To assist with identification of current protocols that may apply to a given patient;

(2) To assist with determining a patient's eligibility for a given protocol;

(3) To provide detailed information on protocols in response to questions from clinic personnel;

(4) To assist with chemotherapy dose selection and attenuation for a given patient;

(5) To provide reminders, at appropriate intervals, of follow-up tests and films required by the protocol in which a given patient is enrolled;

(6) To reason about managing current patients in light of stored data from previous visits of (a) the individual patients (b) the aggregate of all "similar" patients.

Advantages over present paper-based protocol files:

(1) Can be kept readily accessible and up-to-date;

(2) Can provide customized patient-specific calculations and advice not possible with a manual system;

31

(3) May be augmented to provide important additional capabilities once interfaced with a patient data base (e.g., the time-oriented data bank [TOD] already used for retrospective data analysis by the oncology division);

(4) Can provide customized explanations of protocol information and the specific recommendations made by the management system;

(5) Can improve the quality of clinical research by encouraging enrollment of all patients in an appropriate protocol, and assuring that necessary data are obtained to assure uniformity of information on patients in the individual study groups;

(6) Can improve the quality of patient care by:

(6a) Saving time by making protocol information easily available, thus decreasing the waiting time patients must now occasionally sustain while physicians track down necessary protocol information;

(6b) Making certain that important tests are done to screen for potentially serious toxicity of the powerful agents used in cancer chemotherapy.

II.C.   Methods

II.C.1.   Overview

Our general approach to the research will be to emulate the organizational and technical framework used during development of several interdisciplinary computing efforts involving Stanford's Heuristic Programming Project (HPP), of which Prof. Buchanan is co-director. The cohesiveness of project workers has always been facilitated by a weekly group meeting in addition to smaller working sessions at other times. At group meetings both computer science and clinical personnel have opportunities to present their work and give and receive suggestions regarding further efforts. We believe it is important that the physicians and computer scientists get to know each other and their motivations for involvement in the project very well. For example, the computer scientists working on MYCIN have all learned a great deal about infectious diseases, and some have even taken formal courses in microbiology at the medical school. Similarly, the clinicians have been encouraged to understand the program in depth and even to try some programming. We would expect similar relationships to develop among the computer scientists and oncologists working on the proposed research. Only in this way can both computer science and clinical concerns be taken adequately into account during system design and implementation.

In addition to the development of the PMS for the oncology clinic, we anticipate continued research into the basic science issues discussed previously. As has been noted, we have already identified several problems that must be solved before complex reasoning programs such as MYCIN can be made available for clinical use. We also anticipate that work in the oncology domain will uncover new problems, not previously encountered, that may require significant modification or redesign of the EMYCIN formalism. Thus we envision two parallel but highly interrelated efforts:

(1) development of the PMS for the oncology clinic, using EMYCIN and writing new production rules to embody the protocol knowledge that will be needed for consultation sessions;

(2) continued mapping of basic science research, from the core research section of this program, into the oncology problem domain in order to facilitate complex decision making and acceptable consultations in the clinical setting.

II.C.2.   Oncology Protocol Management System

The first year of research on the PMS will be spent developing the program before it is made available in the clinic. Years 2-3 will be devoted to revisions and extensions of the protocol management system in light of initial experience with a knowledge base about oncology. Years 4-5 will be devoted to revisions and extensions of the basic methodology, as well as of the working system, to facilitate use of a clinical data base for patient management in oncology and related disciplines. We expect that the five years will be spent as follows:

(1) We will begin by selecting the 2 or 3 most frequently used oncology protocols (e.g., oat cell carcinoma of the lung, Hodgkin's Disease, non-Hodgkin's lymphoma). The extensive knowledge in these documents will be extracted by the oncologists working closely with those who know the EMYCIN formalism well. Although much of the knowledge can be represented in typical EMYCIN production rules, we anticipate that some of the information may be best contained in alternate representation schemes. We therefore expect that new techniques for interfacing EMYCIN production rules with tabular data or algorithmic structures may be necessary. Most problems that will arise along these lines should develop during codification of the first few protocols; since the protocols all follow a similar structured format, it is unlikely that new problems will arise when the 29th or 30th protocol is being considered.

(2) EMYCIN's knowledge acquisition capabilities remain somewhat rudimentary (see next section), so we expect that most new rules will be explicitly written by members of the research group.

(3) Specific attention will be given to extracting knowledge regarding patient eligibility for a protocol, tests and films needed at various stages of treatment, therapeutic alternatives available, and patient-specific indications for modifying or withholding therapy. We recognize that these are the protocol details that are often most difficult for the oncologists to remember or to extract easily from a lengthy written protocol (an up-to-date copy of which may not even be readily available in the clinic).

(4) Once the knowledge has been codified, we will begin internal testing by interfacing the new production rules and knowledge structures with the EMYCIN program. Of particular interest will be the adequacy of EMYCIN's explanation capabilities when interfaced with this new knowledge base.

(5) Modifications will be made to the EMYCIN system in response to suggestions made by the clinicians working on the project as they gain experience with its capabilities. Of primary concern will be an assurance that the human interface is sufficiently comfortable that the other Division oncologists will be willing to experiment with the system once it is introduced in the clinic.

(6) After these first few protocols are operationally managed by the PMS as described, the system will be introduced in the Oncology Day Care Center. Orientation sessions will be given to the clinic oncologists, and suggestions for further refinements solicited.

(7) The next 3-5 therapy protocols will then be added to the system, with appropriate notification to clinic physicians when a new protocol is available for PMS access.

(8) Based on the experience gathered in codifying the first several protocols, a protocol-entry system with editor will be developed. This should greatly facilitate the entry of the remaining protocols, which we anticipate should be fully codified by the end of year 2.

(9) Anticipating an interface with the TOD system described earlier, plus progress in the basic research that we will be undertaking simultaneously, we will next begin to store patient-related data in TOD format within the PMS. Much of the information in the TOD Databank is also required by the PMS, so there would be minimal if any additional effort required of the PMS user.

(10) Assuming a breakthrough in the representation and management of time-dependent variables, we would anticipate that the PMS capabilities would be greatly augmented by access to patient data stored in TOD format. During Years 4-5 we would attempt to begin the implementation of this kind of interface between TOD and the PMS.

All research described above would occur on a research computer that could not guarantee reliable service to the oncology clinic. We therefore recognize that we cannot initially undertake any tasks crucial to clinic or Division operation. The clinic must be able to continue to function even when our tool is unavailable for scheduling or hardware reasons.

Therefore, when the PMS is ready to progress into a more integral role in clinic operation, we would anticipate, in a separate proposal, the need for a dedicated machine to permit reliable clinic service. We recognize that many of the most interesting and challenging decision making tasks, including those related to the use of symbolic reasoning techniques in conjunction with large databases, can not be made available to clinicians without a dedicated computer, but that this is beyond the scope of the present proposal.

III.   A WORKBENCH FOR KNOWLEDGE REPRESENTATION

III.A.   Objectives of the Research and their Significance

Our primary strategy for conducting our investigations has been to allow the problem to condition the choice of scientific paths to be explored. Projects One and Three, dealing with problems in oncology outpatient consultations and with the clinical laboratory, are the newest examples.

We are also motivated, however, by the importance (to us and others) of generalizing our techniques and systematizing our methodology. This is a normal part of the activity of cumulating the results in our science, in which the experiments we choose to generalize upon are the experimental systems we construct for different domains of knowledge. In Computer Science, one effective method of cumulating our growing understanding is construct software packages that are the working manifestation of what we believe we have come to understand. These packages allow us to transfer yesterday's "experimental technique" into tomorrow's "tool" for accelerating the research.

These packages also allow investigators in other institutions to build rather directly upon the results of our work, thereby amplifying the science as a whole. It is particularly appropriate to cumulate our knowledge as software packages in the SUMEX-AIM community in which the users share the same computer and system.

We have sought to extract from our various projects the uniformities that have general applicability; to eliminate the ad-hoc features that accrue in any large-scale programming effort; and to build helpful "front-end" interfaces that will allow others to couple smoothly to our work.

A number of such packages are beginning to emerge. We propose to continue their development and test; and to merge them appropriately into a larger software system that (for lack of a better term) we refer to as the "knowledge representation workbench".

The Stanford group is fortunate to have the collaboration of the Missouri group to act as a test-and-evaluation site for this workbench concept. It is expected that much of the research of Project Three will be done using the emerging "workbench".

37

We propose the following major objectives:

1.   To develop AI technology as software packages that solve
general classes of problems.

2.   To actively disseminate the technology by publication and
by encouraging pilot projects using the technology.

3.   To apply these packages to medical applications forming
collaborations over time as opportunities arise.

III.B.   Background and rationale

Artificial   intelligence   research   at   the   Heuristic
Programming Project has concentrated on programs having real-
world applications.   Each program has been a case study for
representing and manipulating the task-specific knowledge for an
application.   Feigenbaum [22] has described this case study
approach as essential in building a science for "knowledge
engineering".

Because the cases have been carefully chosen, the
experience from this approach has accumulated. For example, the
GA1 program [53] was developed recently for inferring DNA
structures from enzyme digest data.   This program used the
Generate-and-Test paradigm — in which the combinatoric output of
a complete and canonical generator of possible structures is
limited by pruning rules which use the digest data.   That basic
approach was pioneered by the DENDRAL [11] program ten years ago.
With DENDRAL as an example, the development of this analogous
program was completed in only two months.

This example shows how the accumulation of theory speeds
the development of new AI programs. Significantly, the Heuristic
Programing Project has also accumulated methods — in the form of
software   packages which can   perform specific   symbolic
computations. These packages are the state-of-the-art tools for
applied artificial intelligence. A trained "knowledge engineer"
can combine these packages to create computer programs for new
applications — without having to re-program the solution of
standardized subproblems which have been solved before.

EMYCIN[1] is an example of such a package. It is the domain independent core of the MYCIN [51] program for the diagnosis of infectious diseases. EMYCIN provides a framework for building consultation programs in various domains. It uses a production rule mechanism and backward-chaining control structure during the solution phase and has dialogue facilities for acquiring a production rule knowledge base. An example of an application of EMYCIN is the PUFF system for diagnosing pulmonary function disorder. PUFF was the product of a collaboration with the Pacific Medical Center in San Francisco. the first version of PUFF was built in the following way. One hundred cases, carefully chosen to span the variety of disease states were used to extract 55 rules. The knowledge base was created with EMYCIN and then tested with 150 additional cases. Agreement between PUFF and the human expert was excellent and a later version of PUFF is now in routine use at PMC. The first version of PUFF was created in less than 50 hours of interaction with experts at PMC and with less than 10 man-weeks of effort by the knowledge engineers. Other applications of EMYCIN will be discussed in the Section III.C..

The example shows that methods, in the form of usable computer packages, have . now been developed. These packages reflect the commonalities we now perceive among separate applications. They are the recently available tools of applied artificial intelligence — programs providing practical symbolic methods for common problems.

Our current repertoire of "methods" packages also include the Unit Package, and AGE-1. The EMYCIN program, as discussed above, is based on production rule technology and has been successfully applied to diagnosing pulmonary function disorders and consulting on structural analysis in an engineering application. The Unit Package [52] is based on the so-called "frames" approach and is being applied to experiment planning in molecular genetics. The AGE-1 program is based on the HEARSAY [20] "cooperating knowledge sources" model and is the product of experience with the SU/X and SU/P [43] programs.

New applications are currently being developed for each of these packages. Heiser and Brooks at the University of California at Irvine are using EMYCIN to develop a psychopharmacology consultant, termed HEADMED [34]. Blum [5] has proposed using the Unit Package in a system which will combine statistical methods and artificial intelligence techniques to perform studies on a clinical database. Several other applications have been proposed and are under consideration.

----

[1]The name "EMYCIN" comes from "essential MYCIN", the MYCIN reasoning framework without any domain-specific knowledge.

We propose to continue the development and application of these packages and to develop new ones as results become available from core research.

III.B.1.   Relating the Workbench to Core Research

Over the five year course of this research, there will be a movement of topics from core research into developed packages for the workbench. Our overall strategy has two main thrusts:

1.   To expand the problem solving capabilities of the workbench by developing more sophisticated methods of symbolic reasoning.

2.   To expand the capabilities of existing packages following core research in other topics — knowledge acquisition, knowledge integration, tutoring, and explanation.

This mode of research reflects a bias towards the creation of systems to perform specific tasks. First an approach to problem solving is developed and tested in a task domain. Then research in other topics follows. Three methods of problem solving are discussed in this proposal and elaborated in the following. The simplest of these is a backwards chaining approach — exemplified in EMYCIN — which links together the premises and conclusions of rules to construct a direct line of reasoning. The next level of sophistication in these packages is represented in the AGE-1 which is based on the HEARSAYII [20] architecture. AGE-1 allows (1) both data-driven and goal-driven reasoning and (2) reasoning at different levels of abstraction. This architecture has been used effectively by Stanford researchers in a signal-processing application [43]. Providing other AI capabilities — such as explanation or knowledge acquisition — is more difficult in AGE-1 than in EMYCIN. The next level of sophistication appears in a proposed "planning package" which is expected to grow out of on-going research in the MOLGEN project. This approach to planning formalizes the selection of what to do next as a choice in any of several problem-solving "spaces". The viability of the latter problem-solving method is still being tested and essentially none of the other system capabilities have been developed.

The following is a list of several AI issues discussed in this proposal. These will be explored within some formalisms

already developed by us, EMYCIN, AGE-1, and the Unit Package [2], as well as new formalisms,e.g., the Planning Package as the need arises. The planning package is expected to materialize at the end of some core research which is currently in progress.

Problem Solving
Knowledge Acquisition
Explanation
Tutoring
Knowledge Compiling
Time-Dependence
Meta-Knowledge

III.C.   Methods of procedure

This section describes our plan for creating an integrated collection of well-designed software packages, which can be combined by a knowledge engineer to meet the needs of a specific application. In this section we will show examples of each of the packages and discuss the nature of their applications. We will also discuss the work proposed for further developing the packages.

There is a great deal of overlap in the proposed work among the packages. While the packages reflect different approaches to problem solving and differ in their state of development, analogous lines of research are proposed in each. The EMYCIN package, which is the most developed, uses the the simplest approach to problem solving and has the broadest range of proposed work following several lines of core research. As discussed already in Section III.B.1., similar lines of development are planned later in the grant period for the other packages.

III.C.1.   EMYCIN

The EMYCIN ("Essential MYCIN") project is an attempt to provide a framework for building consultation programs in various domains. It uses the domain-independent components of the MYCIN

-----

[2]The Unit Package is a passive representation package and does not provide any software for problem-solving. It is being used, however, as a representation medium for the Planning Package and can also be used in conjunction with AGE-1.

system, notably the production rule mechanism and backward-chaining control structure. Then for each particular consultation domain, the system builder supplies the rules and parameters of that domain to produce a functioning program. Work on the EMYCIN project is devoted to providing a useful environment for the new system builder, with emphasis on speeding the acquisition and debugging of the knowledge of the new domain.

III.C.1.a. An Example of EMYCIN — The PUFF Application

The PUFF system for the interpretation of laboratory measurements from the pulmonary function laboratory. The EMYCIN system was used as base upon which 60 production rules concerning the presence of pulmonary disease were created. The data from over 100 cases were used to create the rules by the pulmonary physiologist in cooperation with the biomedical engineers who instrumented the laboratory and Stanford computer scientists who had previous experience with the MYCIN program.

Figure 1 shows several rules created during the development of the system. These rules are used to create a complete report including the input measurements, historical information, and the measurement interpretation. Figure 2 shows a copy of this report.

```
IF   0 < DLCO < 80     (DLCO is the measurement of diffusion
                                capacity for Carbon Monoxide)
THEN "Low diffusing capacity indicates loss of alveolar
capillary surface which is "
        IF  70 <= DLCO < 80  THEN "mild"
        IF  60 <= DLCO < 70  THEN "moderate"
        IF   0 <= DLCO < 60  THEN "severe"
```

```
IF The severity of obstructive airways disease of the patient is
      greater than or equal to mild, and
      The degree of diffusion defect of the patient is greater than
      or equal to mild, and
      The total lung capacity measured by the body box (TLCB) is
      greater than 110 percent of predicted,
THEN "The low diffusing capacity, in combination with
obstruction and a high  Total Lung Capacity, would be consistent
with a diagnosis of emphysema."
The subtype of obstructive airways disease is emphysema.
```

Figure 1. Typical PUFF interpretation rules. Conclusions are made for internal system use and for inclusion in the summary.

```
PRESBYTERIAN HOSPITAL OF PMC          DOE JANE          5S2
  CLAY AND BUCHANAN, BOX 7999         P336666.
  SAN FRANCISCO, CA. 94120            DR. SMITH, JOHN
  PULMONARY FUNCTION LAB
WT 56.7 KG,    HT 166 CM,    AGE 58    SEX F
SMOKING 40 PK YRS,CIG 1.0 PK QUIT  0,PIPE  0 QUIT  0,
CIGAR 0 QUIT  0
DYSPNEA-W/MILD-MOD. EXER, COUGH-NO  , SPUTUM-LT 1 TBS, MEDS-YES
REFERRAL DX-CORONARY ARTERY DISEASE , PRE OP
```

```
***********************************************TEST DATE 10-26-78
                        PREDICTED                POST DILATION
                        (+/-SD)   OBSER(%PRED)    OBSER(%PRED)
INSPIR VITAL CAP (IVC)  L    3.1(0.4)  3.0   ( 98)
RESIDUAL VOL     (RV)   L    2.1(0.3)  3.0   (140)     3.5
                                                       (166)
TOTAL LUNG CAP   (TLC)  L    5.2(0.7)  6.0   (116)     6.5
                                                       (125)
RV/TLC                  %    40.       49.            53.
FORCED EXPIR VOL(FEV1)  L    2.6(0.3)  2.1   ( 81)     2.1
                                                       ( 84)
FORCED VITAL CAP (FVC)  L    3.1(0.4)  2.9   ( 96)     3.0
                                                       ( 98)
FEV1/FVC                %    83.       70.            71.
FORCED EXP FLOW 200-1200L/S  4.2(0.8)  4.5            4.4
FORCED EXP FLOW  25-75% L/S  2.9(0.7)  1.5            1.5
FORCED INS FLOW 200-1200L/S  2.9(0.6)  2.9            2.9
AIRWAY RESIST(RAW) (TLC= 6.0) 1.1(0.5) 1.6   (HIGH)   1.4
DF CAP-HGB=14.4    (TLC= 5.3) 25.      17.2  ( 68)
                                             ( 69%IF TLC= 5.2)
************************************************************************
```

INTERPRETATION: Elevated lung volumes indicate
overinflation. In addition, the RV/TLC ratio is increased,
suggesting a mild degree of air trapping. Forced vital capacity
is normal but the FEV1/FVC ratio is reduced, suggesting airway
obstruction of a mild degree. Reduced mid-expiratory flow
indicates mild airway obstruction. Obstruction is indicated
by curvature in the flow-volume loop of a small degree.
Following bronchodilation, the expired flow shows slight
improvement. This is confirmed by the lack of change in
airway resistance. The low diffusing capacity indicates a
loss of alveolar capillary surface, which is moderate.

CONCLUSIONS: The low diffusing capacity, in
combination with obstruction and a high total lung capacity
would be consistent with a diagnosis of emphysema. The
patient's airway obstruction may be caused by smoking.
Discontinuation of smoking should help relieve the symptoms.

PULMONARY FUNCTION DIAGNOSIS:
1. Mild Obstructive Airways Disease.
       Emphysematous type.

                                    Robert Fallat, M.D.
       Figure 2. Sample PUFF Report

III.C.1.b.   Applications of EMYCIN

To date, EMYCIN has been successfully applied at Stanford
to the domains of pulmonary function (PUFF) [37] and structural
analysis (SACON) [3]. [3] EMYCIN is also being applied to clinical
psychopharmacology [34] at the University of California at
Irvine.

III.C.1.c.   Proposed Work for EMYCIN

SYSTEM-BUILDING TOOLS

1)   Acquisition   of   Knowledge   —   Acquire   the

---

[3]SACON (Structural Analysis Consultation): The purpose of
the consultation is to provide advice to a structural engineer
regarding the use of a structural analysis program called MARC.
The MARC program uses finite-element analysis techniques to
simulate the mechanical behavior of objects.   The engineer
typically knows what he wants the MARC program to do, e.g.
examine the behavior of a specific structure under expected
loading conditions, but does not know how the simulation program
should be set up to do it. The MARC program offers a large (and,
to the novice, bewildering) choice of analysis methods, material
properties, and geometries that may be used to model the
structure of interest. The user must learn to select from these
options an appropriate subset that will simulate the correct
physical behavior, preserve the desired accuracy, and minimize
the (typically large) computational cost. The goal of the SACON
program is to bridge this gap, by recommending an analysis
strategy. This advice can then be used to direct the MARC user
in the choice of specific input data, e.g. numerical methods and
material properties.

The performance of the SACON program matches that of a human
consultant for the limited domain of structural analysis problems
that was initially selected.   To bring the SACON program to its
present level of performance, about two man-months of the
expert's time were required to explicate his task as a consultant
and formulate the knowledge base, and about the same amount of
time implementing and testing the rules (this estimate does not
include the necessary time devoted to meetings, problem
formulation, demonstrations and report writing).

framework, vocabulary, and decision rules of the domain from the expert.

2) Rule Checking — Check syntax and semantics of new rules and check for possible conflict with existing rules.

3) Alternative Models for Reasoning under Uncertainty — Provide the system builder with a fixed set of alternative methods for propagating degrees of certainty in the reasoning chains.

4) Time-Dependent Features — Enable the system to make use of parameters whose values change with time.

5) Meta Knowledge — Add capabilities for using meta-rules and other meta-level knowledge.

In addition, we propose extending the power and flexibility of the present system in the following ways:

DOMAIN-INDEPENDENT CONSULTATION SYSTEM

1) Answering Questions — Incorporate question-answering capabilities into the system.

2) Tutoring — Couple the system to a tutoring program to teach the contents of the knowledge base.

Many of these items involve substantial research before we understand the best way to add them to the program or even what, precisely, needs to be added. We present below our best ideas on the approach we will take, but wish to emphasize that the nature of the solution may change as our research progresses.

The products of the research will be presented in scientific papers and in an integrated computer program that can be used by scientists to encode their own knowledge of their domains for reasoning about difficult problems.

45

III.C.1.d.   Acquisition of Knowledge

The preliminary facilities for acquiring knowledge (called TEIRESIAS [Davis76]) developed in the context of the MYCIN application will be incorporated into EMYCIN for use by experts when building any consultation system. This facility will allow an expert to specify the major parameters of a consultation. Then, following a consultation, the system will show the expert the values of these parameters, and ask for verification that they are correct. If the values are not correct, the system will explain to the expert the line of reasoning that led to the incorrect values. This allows the expert to pinpoint an error in the system's rule set, which the expert can then repair by adding, deleting, or modifying rules.

In addition to incorporating the existing rule-acquisition facility, we plan to automate the aquisition of a large portion of the initial knowledge that is required in building a consultation system. The system will prompt an expert through an intermediary for the conceptual framework, vocabulary, and major lines of reasoning of the domain before any rules are entered. The conceptual framework includes the definition and hierarchy of objects or states that will be used to structure the reasoning process (called the "context tree") as well as the attributes and values of these objects that will be used for writing rules. Numerous internal pointers needed for correct associations among concepts will be set up automatically at this time.

Improvements to Teiresias

The TEIRESIAS facility, for interactively debugging the rule base, is most useful when the knowledge base is reasonably well developed and the necessary changes to the rule and parameter base are small. This facility is currently being improved primarily by using the existing question-answering system to explain the system's lines of reasoning [48], and by using a new English parser based on a semantic grammar to understand any rule additions or changes from the expert [8].

An EMYCIN sketchpad As a result of our recent experience eliciting a rule base for structural mechanics [3], we have found it useful to characterize the knowledge acquisition process as occurring in a number of distinct phases.

The first phase corresponds to making initial decisions about the typical advice the consultant will give and the major reasoning steps the consultant will use.

46

This is followed by an extended period of defining parameters and
objects and then, using this initial domain vocabulary,
developing a substantial portion of the rule base. This
process, lasting approximately 2 months in the structural
analysis case, captures enough domain expertise to allow the
consultation system to give advice on the large number of
common cases.

In the final phase, further interactions with the expert tend to
refine and adjust the established rule base, primarily to
handle more obscure or complicated cases.

Future research on knowledge acquisition will explore the
design and implementation of interactive facilities to be used
during the early phases of the knowledge base design. In
particular, methods will be developed for rapidly acquiring and
manipulating definitions of the context tree of objects, their
major parameters, as well as the major problem solving strategies
to be used by the consultant.

During the initial passes at defining objects, the system
would begin to acquire some detail about the actual methods (the
rule sets) that will be used to reason about the major parameters
of the consultation. For each of these parameters the expert
typically knows what major factors and subgoals will be relevant
to concluding the parameter. These factors can be specified by
the expert, but need not be acquired in detail until the system
actually must begin gathering the rules for determining these
important parameters. In this manner, the expert can be free to
concentrate on the more general aspects of the problem solving
process without having to be bothered with the specification of
detail.

Using the EMYCIN sketchpad, the expert and intermediary
would develop and acquire substantial portions of the knowledge
base and an explicit representation of the overall reasoning
strategy that the program will use to advise about the user´s
problem. This framework and knowledge of overall strategy can be
used later to motivate explanations of the system´s lines of
reasoning produced by the question-answering system. We intend
to investigate ways that this knowledge about the major
parameters could be used by TEIRESIAS (during the later phases of
the knowledge acquisition process) to explain how and why a
particular, incorrect conclusion was made.

Rule Checking

While the production rule format permits any executable
LISP expression as the premise or action of a rule, not all LISP
forms make reasonable rules. Common syntactic errors include
misspellings, misplaced arguments, parenthesis errors and
incorrect classification of the rule; such errors generally
result from inaccurately inputting the rule, and if left
undetected, may cause the rule to fail, or even cause runtime
errors. Semantic errors can result if a new rule is inconsistent
with existing rules, or is incomplete, failing to take into
account all the factors necessary for the conclusion.

We plan to do extensive checking of each new rule entered
into the system. We hope thereby to catch most errors at rule
entry time, rather than finding them during later consultation
runs when it is harder to (a) isolate the effects of a faulty
rule and (b) correct any problems which result.

Syntactic checking is fairly straightforward. The rule
checker needs to know about the syntax of each argument to the
predicates which make up a rule. This knowledge exists in the
form of predicate templates, which have long been used by other
parts of the system to "read" rules. The rule checker's use for
them is, in effect, to make sure the rules are "readable". For
example, the template for the predicate SAME is
        (SAME CNTXT PARM VALUE),
for which a typical instance from the infectious disease domain
might be
        (SAME CNTXT IDENT E.COLI).

The rule checker knows from this that a call to SAME should
have three arguments: the first must be a legal "context atom",
i.e., a variable used to select a binding in the context tree,
the second must be a parameter, and the third must be a legal
value for that parameter. If any of these is incorrect, the
error is easily detectable, and in many cases correctable.
Simple spelling errors may be corrected by invoking INTERLISP's
spelling corrector, using an appropriate spelling list; e.g., for
the PARM slot use the list of all parameters, for the VALUE slot
use the list of values legal for the parameter appearing in the
PARM slot. Transposed arguments and spurious extra arguments
(typically a result of parenthesis errors) are also easily
detected by checking against the template.

Another common syntactic error is incorrect classification
of a rule, i.e., specification of what type of context it may
apply to. In many cases it is possible for a rule checker to
completely determine the correctly classification, simply by
observing which parameters appear in the rule and comparing with
the known structure of the context tree. At worst, the checker

could narrow down the possibilities to a small set of nodes of
parallel structure.

More subtle errors arise from fundamental "semantic" errors
in a new rule, and the processing required to detect such errors
is correspondingly more complex. One major type of semantic
error is inconsistency of a new rule with existing rules. One
rule might subsume another, i.e., one premise is implied by
another. For example, with the two rules

         A   -> X
         A & B -> X,

the first subsumes the second. The error here is that if the
second rule succeeds, the first will also, and the information A
is contributing twice to the conclusion X. Our certainty factor
model is predicated on rule premises being independent;
subsumption is a blatant violation of that assumption.

Another possibility is that one rule might contradict
another rule or rules. This is trickier. Certainly the two
rules

         A -> X
         A -> ¬X

contradict each other. But such obvious contradictions are
fairly unlikely; more subtle interactions can occur. For
example, given a set of rules

         A -> B, B -> C
         A -> D, D -> ¬C

it is difficult to determine whether there is a contradiction
except in the special case that all the rules have definite
conclusions (CF=1.0). But if the confidence attached to those
conclusions is less than definite, there may be no direct
contradiction at all, merely conflicting tendencies, perfectly
admissible under our certainty factor model. We   plan to
investigate means of analyzing rules to uncover possible
contradictions, measure how great a conflict may exist, and ways
to determine if the conflict is a real problem.

Another type of semantic error may occur if a rule fails to
take into account all the information relevant to a conclusion.
The system can sometimes detect this by means of rule models,
which currently consist of statistical observations of the
correlation of parameter occurrences in existing rules [15].
These rule models are constructed automatically by reading the
rules. As a typical use, if rules mentioning parameter x usually
also mention parameter y, then the system might request
confirmation of a new rule which considers only x. We plan to
increase the richness of the rule model language, to enable
better semantic checking of the user's rules, especially during
early acquisition phases, when there do not exist sufficient
rules to form useful rule models on purely statistical grounds.

For example, the user might wish to describe in some brief fashion the sort of rules he is about to enter, and the system could then make sure the rules are actually consistent with the user's model.

III.C.1.e. Alternative Models for Reasoning under Uncertainty

The method developed for ranking MYCIN's hypotheses based on measures of certainty is an approximate method. It developed from a pragmatic need for measuring the degree of confirmation of a hypothesis based on several non-independent (partially overlapping) pieces of evidence. The certainty factor (CF) model discussed above is a means of combining single "certainty factors" associated with each inference to arrive at a reasonable measure of how strongly the evidence supports each hypothesis.

It is reasonably simple to understand. However, its main drawback lies in the difficulty of associating a CF with a single rule. Because the rules are not independent, the CFs are also not independent. This means that adding a new rule involves looking at similar rules in order to decide how high the CF ought to be set.

For some experts (or problem areas), CFs seem to be more difficult to use than for others. Thus we propose to offer the system builder a choice of evidence accumulation methods. One of them will be the CF scheme already in use. A second will be the likelihood ratio scheme used in the PROSPECTOR system [18], although that requires storing two measures with every inference: $P[H/E]$ and $P[H/\bar{E}]$.

A third method will be a very simple additive measure with thresholding, as proposed by one of the physicians working with MYCIN. In this model, measures of positive and negative evidence are added and subtracted into a total for each hypothesis, with action taken on the hypotheses in the end that lie above the threshold.

Under other funding we are exploring other relationships between evidence and hypotheses. As measures are found that can be fit to new problem-areas we will find ways of adding them to the set of available confirmation methods. The important point here is to give the system builder a choice of evidence accumulation schemes, any of which can be used in EMYCIN.

Time-Dependent Features

A consultation system built under the current design of
EMYCIN takes a snapshot of the available information about a case
and makes a one-time evaluation of the situation. In cases where
the nature of the diagnosis or repair is strongly dependent on an
understanding of the process of failure over time, this static
approach to the problem is inadequate. No provision is made in
the present system for considering the same case several days
later when more information is available or when the values of
some parameters have changed.

The system also lacks a mechanism for dealing with
parameters whose values vary with time. In many domains, time
considerations may be crucial to the solution of even the
simplest problem. For example, it might be critical to track the
values of various parameters over a period of time, or to check
what value existed at a particular time in the past.

In order to increase the number of domains in which EMYCIN
systems will be useful, we plan to add two new features. The
first is a "restart" mechanism that will allow a user to run a
follow-up consultation on a stored case, adding information that
has become available since the original consultation, and
correcting old answers that are no longer accurate. The second
is to expand the syntax and semantics of rules to deal with
values of parameters changing over time.

Follow-up Consultations

The builder of an EMYCIN system should be able to specify
which parameters are likely to change for a given case from one
consultation to the next. In a follow-up consultation, the
system should summarize its knowledge of the case and do the
following three things:

    1) ask whether new information is available
for any of the parameters which are subject to
change, and prompt for the new answers;

    2) ask whether values are known for any of
the parameters whose values were UNKNOWN at the
time of the previous consultation, and prompt for
the new answers;

3) allow the user to specify changes which may have occurred in the values of any other parameters (viz., those which do not usually change).

Extending the Rule Syntax and Semantics to Deal with Time Relations

The builder of an EMYCIN system should be asked to classify parameters according to their stability over time. A possible classification scheme is shown below.

1) Constant – value is always the same (e.g., Name and Sex of medical patients)

2) Regularly changing – new value is available at regular intervals; there will be several values stored for the parameter, each with a time (e.g., barometric pressure at a certain city)

3) Gradually refined – value is likely to change over time, from unknown to uncertain to definite (e.g., Identity of an organism growing on a culture plate)

Parameters of the first type are the typical case that EMYCIN now handles. For the second type, a time must be kept with each value–CF pair. The third type of parameter will typically change from one consultation to the next, and previous values will be discarded as new information becomes available.

New PREMISE and ACTION functions must be defined so that EMYCIN rules can handle time–varying parameters. Functions will be needed to test and conclude (a) the value of a parameter at a given time, (b) the duration of a particular condition (e.g., it has been raining for three hours), and (c) trends in the values of numeric parameters (e.g., the volume of water in the tank has increased within the last hour). As we test EMYCIN in different domains, we may discover other types of tests and conclusions that must be made on time–dependent parameters.

Add Capabilities for Using Meta-Rules and other Meta-Level Knowledge

Our preliminary research with meta-level knowledge [15] as

well as our preliminary experience with the GUIDON tutorial program has shown the importance of acquiring, using and teaching structural and strategic meta-knowledge, as well as the domain rules. Structural meta-knowledge provides a framework that sets the context for domain rules, and in tutoring helps make the rules memorable to a student. It might include patterns and principles that are made specific by groups of rules. Strategic meta-knowledge constitutes planning knowledge for using the rules to solve different problems [10]. This meta-knowledge is written as meta-rules and takes the form of diagnostic reasoning strategies and domain-dependent approaches for efficient consideration of a case.

In our work with EMYCIN, we will explore various kinds of structural and strategic meta-knowledge that is appropriate to the production rule representation and useful for explaining decisions made by the program (to a consultation user or a student). We will start by implementing in EMYCIN the capabilities for using the meta-level knowledge described by Davis: meta-rules to be used for pruning and reordering the object-level rules, and meta-level models of rule sets that aid in debugging (and tutoring) the domain knowledge.

Experience with EMYCIN programs like HEADMED and PUFF will provide us with particularly useful case studies of possible forms of meta-knowledge.

Incorporating Question-Answering Facilities into the System

In order to make the questions-answering facility available to an EMYCIN consultation system, the system must be provided with a dictionary of synonyms and a list of definitions of the important concepts in the its domain of expertise. The dictionary will contain common synonyms in the domain, pointers between English words and parameters, and common phrases in the domain that can be given a single specified meaning.

We will provide a facility for automatically constructing a dictionary from the parameters in the knowledge base. The system builder will also be able to add synonyms and fill in parts of the dictionary that cannot be created automatically. This should provide all the information necessary for answering standard questions about the consultation system. The kinds of questions that the system will be able to answer are:

1) the value of a parameter

53

2) how a parameter was used or concluded in the consultation

3) how a parameter is used or concluded in general

4) how a rule was used in the consultation

5) why a question was asked during the consultation

6) the translation (into English) of a rule

7) the definition of a concept

These question types will be recognized in a variety of forms. For example, all of the following will be taken to be equivalent ways of asking for the value of a parameter

1) What is the value of X?

2) Is Y the value of X?

3) What is X?

4) Do you know what X is?

The major benefits of providing these capabilities are that the user of a consultation system can understand the reasoning and the designer of the system can find the sources of reasoning errors.

Coupling a Tutorial System to EMYCIN

Work on the idea of automatic "Transfer of Expertise" from a human expert to a program [22], [15] has led to important advances in the representation of knowledge within the program. These advances have allowed the systems to explain their reasoning process to users, thus providing the basis for a tutorial program. We have been building an intelligent computer aided instruction (ICAI) program [12] that guides a subject through problems in a complex domain with the goal of transferring the system's knowledge of the domain to the student.

Current ICAI techniques like planning the discourse, modelling the student, and teaching problem solving strategies all take a natural form in our system. In turn, the system serves as an excellent environment for experimenting with unsolved problems in the design of computer-based tutoring.

We have demonstrated the feasibility of using the MYCIN knowledge base for teaching as well as for consultation, and this aspect of our research will be continuing during the grant period under separate funding[4].

We have not yet demonstrated the generality of the tutorial program, GUIDON, in other domains; but we have meticulously avoided introducing any domain-specific knowledge into GUIDON's control structure and teaching strategies. We believe that its design is as general as MYCIN's. Thus, all that is needed for tutoring in another domain will be (a) domain rules for EMYCIN to use on cases which GUIDON can discuss and (b) domain specific meta-level knowledge that would be useful for teaching these rules. Moreover, we must keep the tutoring strategies of GUIDON coupled to the representation of EMYCIN systems that we wish to tutor.

III.C.2.   AGE-1

The basic idea behind AGE-1 is to generalize the ideas found in specific problem-solving systems and make them available in a package — hence the name AGE, for "Attempt to GEneralize". AGE-1 takes an active role in assisting a knowledge engineer in constructing a performance system. The specific model that is incorporated in AGE-1 — the "cooperating knowledge sources model" — was pioneered in the HEARSAYII system ([20], [33]) for speech understanding. It was further developed by Stanford researchers in two data interpretation problems — SU/X and SU/P (otherwise known as HASP and CRYSALIS) [43].

III.C.2.a.   Examples from AGE-1

The CRYSALIS program [19] is a knowledge-based program being developed in collaboration with the University of California at San Diego. Its task is to infer protein structure from X-ray crystallography data. This program was developed in

---

close collaboration with the AGE group at Stanford and has been using a very similar problem-solving model. Currently the top-level of CRYSALIS is being rewritten using the AGE-1 package. Examples from the CRYSALIS program are used below to illustrate the problem-solving model in AGE-1.

The Problem-Solving Model

AGE-1 uses a uniform multi-level data structure, termed the "blackboard", to hold the status of the system. In CRYSALIS, the blackboard is used to hold various crystallographic data and structural hypotheses. Separate hierarchically organized panels of the blackboard correspond to "electron-density" space and "protein-model" space. These correspond roughly to data space and hypothesis space except that the electron density space has two levels of hypotheses above the electron density data. The protein-model space describes the three-dimensional structure of the protein at different levels of abstraction from the atomic level to the large-scale structural features like "beta-sheets".

Skeletal Level (backbone — graph of density nodes)	Stereotypic Level (helices, beta-sheets)
Nodal Level (high intensity points)	Superatomic Level (Side chains, proline)
Parametric Level (electron density data)	Atomic Level (C,N,Fe etc.)

| Electron Density Space | Protein Model Space |

A set of procedures termed knowledge sources (KSs) are used to form and link the hypotheses on these panels. In the CRYSALIS application, these knowledge sources include such domain specific operations as skeletonization, helix identification, sidechain identification, bond rotation, sequence identification, cofactor identification, and heavy atom identification. The knowledge sources are expressed as production rules. AGE-1 provides a framework for coordinating the activity of the KSs mixing goal-driven and data-driven reasoning as it searches for solutions.

If the KSs had been perfect, the coordination could have be

directed in a goal-driven manner analogous to the production rules in EMYCIN. However, because of gaps in the theory and implementation of the individual KSs and noise in the data, they are individually incomplete and errorful. Like the HEARSAYII system, AGE-1 uses an algorithm — a version of the hypothesize and test paradigm — which emphasizes cooperation (to help with incompleteness) and cross-checking (to help with errorfulness). During the hypothesize part of the cycle, a KS can add a hypothesis to the blackboard; during the test part of the cycle, a KS can change the rating of a hypothesis in the blackboard. This process terminates when a consistent hypothesis is generated satisfying the requirements of the overall solution or when knowledge is exhausted.

In AGE-1, the hypothesize-and-test paradigm is formalized as a control structure with three levels. The first level is the hypothesis-formation level. KSs on this level make changes to the blackboard panels. In the hypothesize and test paradigm, they put hypotheses on the blackboard and test the hypotheses of other KSs. A rating is associated with each hypothesis to store the overall judgment. Immediately above the hypothesis-formation level is the KS-activation level which contains two KSs. The KSs are called the "event-driver" and the "expectation-driver" and correspond to data-driven and goal-driven policies for activating KSs on the first level. The highest level of KSs is called the strategy level. This level must decide (1) how close the system is to a solution, (2) how well the KSs on the second level are performing and (3) when and where to redirect the focus-of-attention in the data space. KSs on this level can invoke KSs on the second level.

This problem-solving method is more complex and more general than the backward-chaining approach used in EMYCIN. It is designed to tolerate errorfulness in the data and in the KSs and allows the inferences to be run opportunistically in either direction. It also allows the inferences to be run at several levels of abstraction.

Using AGE-1 to Build a Knowledge-based System

The purpose of the AGE-1 system is to assist a computer scientist at building a problem-solving system. AGE-1 is intended to speed up process task when the task domain can be cast in the model of cooperating knowledge sources. To this end, AGE-1 has several software subsystems — a "TUTOR" subsystem and several knowledge acquisition subsystems.

The TUTOR is a module for the unfamiliar user which helps

him create an application program. It guides the user through a top-down design of his system by presenting him with a list of topics and subtopics at each level. Canned text is available for explaining the choices at each level. A "browse" option is available for random perusal of the topics and subtopics.

Knowledge about the parameters of the application program is acquired by the DESIGN subsystem. The DESIGN subsystem provides the user with choices at each phase of the construction of the application program. This construction involves choices for hypothesis structure, rule acquisition, goals, and expectations. Thus, the domain dependent particulars for each of the components of the application program are asked about in turn. For example, the following items must be acquired for each KS

1.   preconditions

2.   inference levels

3.   links

4.   hit strategy

5.   local variable bindings

The acquisition of each of these items is further broken into the most primitive elements. The DESIGN module has a "guided" approach for the novice and an "unguided" approach in which an expert calls for the knowledge acquisition functions quickly and directly.

III.C.2.b.   Applications of AGE-1

The CRYSALIS example illustrates the most comprehensive application of AGE-1. AGE-1 has also been used on an experimental basis to create a version of PUFF Section III.C.1.b. and on some cryptography problems (simple code-breaking). These applications have been used for testing the tutorial and knowledge acquisition components of AGE-1.

III.C.2.c.   Proposed Work for AGE-1

In the current version of AGE-1, the DESIGN module provides choices and explains them with canned text. AGE-1 does not build up its own knowledge of the user's application — only a knowledge of the design choices that the user makes. It does not make inferences about the relationships between design choices — so that it does not infer choices for the user even when one set of choices implies another set.

We plan to move toward a system where AGE-1 will ask the user about the domain and play a more active role in making the design decisions. This means that AGE-1 must have a model of "how to build a system" and that we must encapsulate the reasons behind the design choices. Our plan is to begin to capture this information in the form of production rules which relate the form of the domain knowledge to the design choices of AGE-1 to a prediction of the performance consequences in the application program being built.

Accompanying this effort we would like to begin construction of two explanation subsystems — one for explaining the activity in the design phase and one for explaining performance of the application system. We expect to build on the explanation work in the EMYCIN system for this.

In the long term, we also plan some work on knowledge compiling. Our plans for this in the EMYCIN system have already been discussed. There is some experience in compiling the knowledge of a cooperating knowledge source system — notably the HARPY [39] system which can be seen as a "compiled" approach to the task performed by HEARSAYII. Much more work is needed before this could be done automatically.

III.C.3.   The Unit Package

The Unit Package is a frame-structured representation system developed as a tool for building knowledge bases in the MOLGEN project. Unlike EMYCIN and AGE-1, the Unit Package provides no problem-solving framework. However, the Unit Package can be used as a passive representational medium in conjunction with specific problem-solving approaches. Two approaches to experiment planning are being developed in this way as part of research in the MOLGEN project. The Unit Package is also accessible from within the AGE-1 package. The Unit Package builds on a substantial amount of work (both here and elsewhere)

on frame-structured languages. A comprehensive description of this work is available as a technical report [52] which is included with this proposal.

Knowledge in the Unit Package is organized in a semantic network of nodes and links. Following other work on frames [42], the nodes are called "units" [6] and the links are called slots. The major software components of the Unit Package are (1) an interactive editor for adding new information or modifying existing information, (2) a set of routines for matching and manipulating descriptions, and (3) a set of access functions which maintain network relations (such as inheritance of properties) and provide an extended address space to hold the semantic network.

III.C.3.a.   Examples from the Units Package

The Unit Package is a fairly extensive set of software for defining the symbolic entities of a domain. It provides a number of conventions and methods for defining standard kinds of relationships between the symbols.

There are three main steps building a knowledge base for a domain with the Unit Package. The typical user of the Unit Package is a computer scientist, although four geneticists on the MOLGEN project routinely use the Unit Package. The main steps are using the interactive editor are as follows.

(1)   Define the symbols of the domain. These symbols take the form of units as illustrated below.

(2)   Define the operations which manipulate these symbols. Operations are procedural knowledge in the form of production rules or LISP functions.

(3)   Define an approach for problem solving.
The steps are not necessarily performed in this order or by one person. In an evolving knowledge base, the user uses the editor both to create new symbols and to modify old ones as his understanding improves. The expertise to define all of these things may be spread over several people working on a common knowledge base.

"Specialization" is a relation which is indicated by a user when he defines a symbol. It is used to indicate subclasses among concepts — e.g., the unit for the restriction enzyme Eco Rl is a specialization of the unit for general restriction enzymes which is a specialization of the unit for endonuclease which is a specialization for the unit for nuclease and so on. General properties of a class are inherited by its specializations. This is formalized in part by having descriptions in slots of those units that correspond to classes. These descriptions delineate legal values for the corresponding slots in specializations of the class. Descriptions can be progressively tightened as one proceeds down a specialization hierarchy. This feature makes the process of specialization correspond to the addition of non-contradictory new knowledge to units. A specialization (or generalization) hierarchy of concepts from a molecular genetics knowledge base is illustrated below.

```
LAB-OBJECT
        ANTIBIOTIC
                AMINOGLYCOSIDE
                        KANAMYCIN
                        NEOMYCIN
                BETA-LACTAM
                        AMPICILLIN
                        ...
        GENE
                APR
                CMR
                ...
        ENZYME
                LIGASE
                ...
                NUCLEASE
                        ENDONUCLEASE
                                RESTRICTION-ENZYME
                                        ALU1
                                        ASU1
                                        ...
```

          Symbols in the Unit Package are
       organized in a generalization hierarchy.
       This hierarchy indicates "inheritance paths"
       by which symbols acquire the attributes of
       their generalizations.

     Each of the symbols in a knowledge base is defined in terms of "slots". A unit corresponds approximately to a property list

except that (1) the structure of a slot has several explicit
fields for information about such things as modes of inheritance
and datatype and whether the value is stored or computed[5] and (2)
the value of a slot can be a description of a value. The
following figure illustrates two units of different complexity.

```
NAME:               Endonuclease
DOCUMENTATION:      A nuclease that cuts internally in a
                    DNA structure.
SITE-TYPE:          One of (MONO, STICKY-HEXA, FLUSH-HEXA,
                    PENTA, STICKY-TETRA, FLUSH-TETRA)
3'-END:             One of (P, OH)
5'-END:             One of (P, OH)
MODE:               One of (Precessive, Non-precessive)
OPTIMAL-PH:         RANGE (0 14)
  ...                 ...
```

```
NAME:               Rat-Insulin-Problem
DOCUMENTATION:      This unit gives the parameters of an experiment
                    for cloning the gene for rat-insulin.
GENE:               RAT-INSULIN
GENE-PRECURSOR:     RAT-INSULIN-RNA
ORGANISM:           A Bacterium
  Default:          E.COLI
VECTOR:             A Vector
GOAL:               A Lab-goal with
                      STATE = A Culture with
                              ORGANISMS = A Bacterium with
                                  EXOSOMES = A Vector with
                                      HAS-GENES = RAT-INSULIN
                    CONDS = (PURE? ORGANISMS)
```

Two units from a MOLGEN knowledge base.
Each unit is organized as a list of slots.
The slots are filled with values or
descriptions of values. These units are
examples of "symbols" from the molecular
genetics domain.

While the Unit Package is not a problem-solving program, it
does provide a large number of routines for creating, modifying,
and matching units in a knowledge base. These routines are
called by problem-solving programs in the MOLGEN project which
are currently being tested. Some of the built-in features —
such as the generalization hierarchy and symbolic descriptions —
seem to be especially useful for problem-solvers that work with

---

[5]See the technical report for details.

abstractions. For a discussion of other features of the Unit Package — such as the various modes of inheritance, set notation, or the attachment of procedural knowledge — the reader is referred to the enclosed technical report.

III.C.3.b.   Applications of the Units Package

MOLGEN — Planning Experiments in Molecular Genetics

Molecular genetics is a rich and rapidly growing science. Several aspects of molecular genetics make it attractive as a task domain for artificial intelligence. It is a young science and new techniques and ideas are developed regularly. This makes it attractive for studying the process of discovery ([38], [23]). It is a laboratory science and experiments are clearly defined in terms of laboratory steps and results. This makes it attractive for studying the processes of planning and plan debugging. Finally, many kinds of knowledge are used in molecular genetics. This motivates work on representation in the Unit Package.

Planning research in MOLGEN has focused on two broad classes of experiments — structural synthesis and structural analysis. The synthesis experiments use various laboratory techniques to build DNA structures. Analysis experiments use various laboratory techniques to identify an unknown structure. An analyst seeks to discriminate between competing hypotheses for the structure of a sample.

Other Applications

In the past few months, several other projects have begun to use the Unit Package as a representational medium. Dr. Blum [5] is using it in an application which will combine statistical methods and AI methods for performing studies on a clinical data bank at Stanford. The Unit Package is being used to represent a set of medical models to permit a more sophisticated interpretation of patient record data in the data base than is possible using statistical methods alone.

The Unit Package is also being used in a mathematical application at Stanford and is being tested for a planning application at the RAND corporation. Other applications are expected over the course of this grant period.

III.C.3.c.   Proposed Work in the Units Package

The proposed work on the Unit Package may be divided into two main categories — representational work and research-related work. Barring surprises from the emerging applications of the Unit Package, most of the work on representational machinery is finished. There are a few outstanding tasks such as (1) generalizing the concept hierarchy to be a concept graph so that units can have more than one generalization and (2) providing some more flexible forms of inheritance. Since the Unit Package became operational in June 1977, the rate of change to the system itself has slowed dramatically. This reflects the need for a stable system for development of applications and the fact that the Unit Package has found an important niche for the applications in the Heuristic Programming Project.

This standstill in development also reflects the current interests of the research group — which is to work on the problem-solving applications of the Unit Package. A great deal more development will become important as this work is completed. For example, the Unit Package provides a substantially richer descriptive language for concepts than is available in MYCIN or EMYCIN. It lacks, however, substantial facilities for knowledge acquisition — beyond a simple interactive editor. As applications of the Unit Package develop, an increased need for a stronger user interface is expected — incorporating such things as the natural language interface (BAOBAB [8]).

Another line of development is the development of standard relationships which appear in many domains. The Unit Package currently provides only a very small set of built-in relationships — such as generalization and specialization — which are utilized by the semantic network processing functions. Creating additional relationships is part of the knowledge-engineering task of applying the Unit Package to a task domain. Some of these relationships — such as "part-of" or "abstraction-of" — seem to appear in many domains. To the extent that these relationships have general utility and can be standardized, they will be made part of the initial knowledge base for new applications — thus expanding the apparent power of the Unit Package and reducing the effort of starting new applications.

III.C.4.   Long Term Work and New Packages

The development of packages over the next five years will be opportunistic — relying on the most usable results from core research in artificial intelligence. Thus, while the following

ideas indicate only our best current ideas for continued
development.

III.C.4.a.   Planning Package

One of the areas in which we see future work is in the
general area of planning. The artificial intelligence research
on this problem is currently being performed in the domain of
experiment planning in molecular genetics. Some interesting
ideas are just beginning to emerge from this work which, if
successful, could become the basis of a,"planning package".

This research is investigating the viability of a new
approach to planning called "orthogonal planning". The thrust of
this approach is to take the elements of a planning out of a
"planning algorithm" and put them into explicit "planning
spaces". Explicit planning operations such as refinement
(mapping from abstract to specific) and evaluation and subgoal
proposing are expressed as operators in a planning space.
Different combinations of these operators can be arranged to
create top-down (goal-driven) planning, bottom-up (opportunistic)
planning, and various hybrid methods. The planning research
seeks to find general methods for deciding when to apply these
different planning operators in order to plan flexibly and
effectively. Currently ten planning operations have been
formalized in the planning space and four strategic operations
have been formalized in a overseeing "strategy space". This
approach is being tested in the domain of experiment planning in
molecular genetics and uses the Unit Package for representing the
symbols and operations in all of the spaces.

III.C.4.b.   Time-Oriented   Knowledge   Representation
Package

One important topic in computer-based diagnosis and therapy
programs is the representation of knowledge about situations that
are changing over time. Most current programs have concentrated
on the interpretation of a single instance in the course of the
patient's disease process. As the patient status changes over
time, a program must be able to modify its representation to
conform to the new situation. The ability to represent trends in
the health of the patient is an important part of the diagnostic
process.

Creation of a package that supports the representation of

changes over time will be important for applications based on clinical data bases. These data bases typically contain the results of a variety of tests which were administered at each patient visit to the clinic. The problem of interpretation of updated test results has also come up in each of our current applications, for example, initially negative culture results that grow out a particular pathogen after several days in our infectious disease program or the comparison of new pulmonary test results with the previous findings. No general purpose approach has been incorporated into these programs.

A program for a particular dynamic clinical setting - interpreting measurements from the intensive care unit has been developed at the Heuristic Programming Project. That program, named the Ventilator Manager (VM) [21], is able to evaluate a stream of thirty measurements provided on a 2 - 10 minute basis by a computer-based physiological monitoring system. The system: (1) provides a summary of the patient physiological status appropriate for the clinician; (2) recognizes untoward events in the patient/machine system and provides suggestions for corrective action; (3) suggests adjustments to ventilatory therapy based on long-term assessment of the patient status and therapeutic goals; (4) detects possible measurement errors; and, (5) maintains a set of patient specific expectations and goals for future evaluation.

Removing the the basic assumption about the regularity of the changes in the ICU setting is the major area of research in the development of this package. A typical problem is the interpretations of a series of test values that are higher than normal over several testing instances. Specialized knowledge about the typical rate of change of the underlying disease process is necessary to determine whether these values represent a trend.

The representation of dynamic settings also requires a model of the stages of the disease and treatment process that best characterize the clinical status of the patient. Often a particular value of a measurement takes on entirely different interpretations based on the current context. For example, the meaning of critical measurements one hour after surgery compared to the same measurement after three days of recovery. A rudimentary model of this type based on various therapeutic regimens is built into the ICU measurement interpretation system. Additional work in required in the generalization of this type of modeling process.

Codification and Use of Medical Knowledge
from Clincial Laboratories

ADMINISTRATIVE INFORMATION ONLY

1. TITLE OF PROPOSAL *(Do not exceed 53 typewriter spaces)*

Clinical Laboratory Expert Project

2. PRINCIPAL INVESTIGATOR	3. DATES OF ENTIRE PROPOSED PROJECT PERIOD *(This application.*	
2A. NAME *(Last, First, Initial)* Lindberg, Donald A. B.	**FROM** ▮▮▮▮▮▮▮▮	**THROUGH** May 31, 1984
2B. TITLE OF POSITION Director, Information Science Group Director, Health Care Technology Center	**4.** TOTAL DIRECT COSTS RE- QUESTED ▮▮▮▮▮ D IN ITE▮	**5. DIRECT COSTS REQUESTED** FOR FIRST 12-MONTH PERIOD
2C. MAILING ADDRESS *(Street, City, State, Zip Code)* University of Missouri 605 Lewis Hall Columbia, Mo. 65211	**6. PERFORM▮▮** ▮▮▮▮▮ *nstructions)* Universi▮▮▮▮▮▮▮▮ri Columbia,▮▮▮▮▮ 5211	
2D. DEGREE M.D. **2E** ▮▮▮▮▮▮	Stanford University Stanford, California	
2F. TELE-PHONE DATA Area Code **314** TELEPHONE NUMBER AND EXTENSION 882-6966		
2G. DEPARTMENT, SERVICE, LABORATORY OR EQUIVALENT *(See Instructions)* Health Care Technology Center		
2H. MAJOR SUBDIVISION *(See Instructions)* Graduate School		

7. Research Involving Human Subjects *(See Instructions)* A. ☒ NO B. ☐ YES Approved: C. ☐ YES – Pending Review Date	8. Inventions *(Renewal Applicants Only · See Instructions)* A. ☒ NO B. ☐ YES – Not previously reported C. ☐ YES – Previously reported

TO BE COMPLETED BY RESPONSIBLE ADMINISTRATIVE AUTHORITY *(Items 8 through 13 and 15B)*

9. APPLICANT ORGANIZATION(S) *(See Instructions)* The Curators of the University of Missouri 215 University Hall Columbia, Mo. 65211	11. TYPE OF ORGANIZATION *(Check applicable item)* ☐ FEDERAL ☒ STATE ☐ LOCAL ☐ OTHER *(Specify)* University
	12. NAME, TITLE, ADDRESS, AND TELEPHONE NUMBER OF OFFICIAL IN BUSINESS OFFICE WHO SHOULD ALSO BE NOTIFIED IF AN AWARD IS MADE H. Kent Shelton Asst. Vice President Financial Services 215 University Hall Columbia, Mo. 65211
10. NAME, TITLE, AND TELEPHONE NUMBER OF OFFICIAL(S) SIGNING FOR APPLICANT ORGANIZATION(S) H. Kent Shelton Asst. Vice President Financial Services Telephone Number (s) 314-882-3512	Telephone Number 314-882-3512
	13. IDENTIFY ORGANIZATIONAL COMPONENT TO RECEIVE CREDIT FOR INSTITUTIONAL GRANT PURPOSES *(See Instructions)* Graduate School
	14. ENTITY NUMBER (Formerly PHS Account Number) 43-6003859

PROJECT 3. Codification and Use of Medical Knowledge from Clinical Laboratories

ADMINISTRATIVE INFORMATION ONLY

RESEARCH OBJECTIVES

NAME AND ADDRESS OF APPLICANT ORGANIZATION

University of Missouri-Columbia

NAME, SOCIAL SECURITY NUMBER, OFFICIAL TITLE, AND DEPARTMENT OF ALL PROFESSIONAL PERSONNEL ENGAGED ON PROJECT, BEGINNING WITH PRINCIPAL INVESTIGATOR

Donald A. B. Lindberg, M.D., ███████████, Director, Health Care Technology Center and
  Information Science Group, Professor of Pathology
Robert Abercrombie, Ph.D., ███████ Post Doctoral Fellow, Information Science Group
Paul Blackwell, Ph.D., ███████ Professor of Computer Science
Lamont Gaston, M.D., ███████ Professor of Pathology
Lawrence Kingsland, ███████ Senior Electronics Technician, Information Science Group
W. B. Stewart, M.D., ███████, Professor of Pathology, Director of Laboratories
Henry Taylor, M.D., ███████ Professor of Pathology
John Townsend, M.D., ███████ Professor and Chairman, Department of Pathology

TITLE OF PROJECT John Yio Ph.D., 227 68 0029, Post Doctoral Fellow, Information Science Gro[
Clinical Laboratory Expert Project

USE THIS SPACE TO ABSTRACT YOUR PROPOSED RESEARCH. OUTLINE OBJECTIVES AND METHODS. UNDERSCORE THE KEY WORDS (NOT TO EXCEED 10) IN YOUR ABSTRACT.

A. Objectives

  1. To represent within a computer-based information system the knowledge and procedures of the clinical laboratory expert.
  2. To determine how to implement this information system such that benefits result to the clinical laboratory service which are measurable in terms of:
     a. Increased quality of laboratory determinations
     b. Reduced costs to the laboratory and/or the institution
     c. Increased access to pertinent information by laboratory data providers and users.
  3. To determine how to interface this information system with the hospital and clinic services such that benefits result in actual patient care. We propose to seek "process" measures rather than "outcome" measures.
  4. Using this operational testbed to shed light upon certain important questions basic to artificial intelligence in medicine research.

  These objectives will be pursued by construction of a knowledge representation system for the domain of the clinical laboratory expert. Subject matter expertise will be provided by directors of the clinical laboratories of the University of Missouri Medical Center. Fundamental artificial intelligence methodology and specialized computational facilities will be provided by the SUMEX Laboratory and the Department of Computer Science at Stanford University. Management and interfacing of the project and site-testing will be provided by the Health Care Technology Center at the University of Missouri-Columbia.

PROJECT 3:   The Clinical Laboratory Expert Project

III. A.   Objectives

1. To represent within a computer-based information system the knowledge and procedures of the clinical laboratory expert.

2. To determine how to implement this information system such that benefits result to the clinical laboratory service which are measurable in terms of:

(a) Increased quality of laboratory determinations

(b) Reduced costs to the laboratory and/or the institution

(c) Increased access to pertinent information by laboratory data providers and users.

3. To determine how to interface this information system with the hospital and clinic services such that benefits result in actual patient care. We propose to seek "process" measures rather than "outcome" measures.

4. To seek through this operational testbed to shed light upon certain important questions basic to artificial intelligence in medicine research. These include the following:

(a) How best to retain the power of symbolic representations traditional to AI techniques while at the same time obtaining the benefits of the numerical methods which are traditional to fields such as laboratory management?

(b) How best to set up an information system so as to accommodate to the endless stream of changes which occur in the operating environment of a system such as the clinical laboratory?

(c) How to improve, and hopefully optimize, the interface

of the knowledge engineer and the subject matter expert, in
this case the clinical laboratory expert?

III.B.   Background and Rational

Use of artificial intelligence techniques, especially the recent
focus on formal representation of the knowledge of experts, is the latest
and most promising of applications of the computer to medicine. It is
already clear that the techniques are powerful and that the proof-of-
concept and feasibility phases of medical applications have been success-
fully passed. This technique has been shown feasible in the areas of
infectious disease (Shortliffe et al., 1973), glaucoma management
(Weiss, Kulikowski, Safir, 1978), patient present illness (Pauker,
Gorry, Kassirer, Schwartz, 1976), and in the general differential
diagnosis in internal medicine (Lawrence, 1978). In many ways the AI
techniques are still in development, but the real question remains:
in what areas of medicine are they most usefully going to be employed?
Some raise the question, in which areas would such techniques even
be accepted?

The clinical laboratories offer the very best application sites
for exploring AI techniques as a basis for biomedical information
systems. The following observations support this contention:

1. The clinical laboratories were the first sites for
successful implementation of computer-based information
systems of any kind (Hicks, 1969; Lindberg, 1965, O'Kane,
Haluska, 1977).

2. There are a host of current computer systems already
disseminated in this field which form a basis for advanced
technological developments.

70

3. Clinical laboratory services constitute a major part

of hospital expenses (estimates vary from 25-40%).

4. Clinical laboratories, for the most part, are

administered by professional medical personnel who have

training in technological matters, including hardware and

information systems, and who therefore are likely to be

receptive to advances in this kind of methodology.

5. There is an expertise in clinical laboratory operation

and interpretation which is recognized by medical specialty

training.

6. Knowledge in this field is plentiful; and expertise

takes the form of a multitude of tiny empirical pieces of

information, which await unification into an overall

information framework. This situation is compatible with the

way in which formal knowledge systems have been built for

other AI applications.

7. On the other hand, the field does offer an advantage

in another (almost counter) sense: namely, that there are

true and realistic models of the basic data generating

sources. For example, one knows quite surely that impedance

transients in a Coulter Counter are caused by particles,

and that these particles are (for the most part) erythrocytes.

Likewise, the concept of "serum electrolytes" is known

to have a solid basis: namely, that there are actual,

immutable ions of sodium, potassium, chloride, and bicarbonate

(and CO_2) within the serum. Furthermore, chemical laws

describe the relationship between many blood constituents.

Curiously, the chemical laws are not used ordinarily as the

basis of laboratory management, and only partially as a

basis for test interpretation and subsequent patient

management. The chemical laws and the physical models

are, however, a potential advantage in building advanced

information systems.

8. The clinical laboratory offers a setting which is

receptive to and safe for development of new information

systems, yet which also offers a home base for extension

out toward the more purely clinical setting. The meeting

ground of the two is clear: it is the interpretation of

the results of laboratory measurements.

For these reasons, we feel that clinical laboratories are in

general a potentially fruitful place for AI in medicine applications.

There are reasons which make us think that the particular

laboratories and group at the University of Missouri are a good

choice among those institutions with excellent clinical laboratory

programs.

1. The school has a long history in lab system developments.

The first automated lab system in the country was built here

In 1962 and has operated continuously since then.

2. The system incorporates all clinical laboratories and all

test results.

3. These results are in computer processible form, indeed

are reported through the computer systems. Consequently test

data Is accessible.

4. Experts in clinical laboratory medicine are members of the

team who propose to build the Clinical Laboratory Expert system.

5. The project is sponsored by the Health Care Technology

Center, which has ample experience and capability in the
management and conduct of multi-disciplinary technical
projects. The Center management review of all projects
includes participation of an evaluation team with members from
operations research, medical sociology, economics, health
services management, and medicine.

6. Most important of all, we have a plan to accomplish the
system building, and we have predecessor systems to build
on and to compare with.

III.C.   Methods of Procedure

We propose to grow the information system beginning with a
nidus or model system and to expand the scope of the system by
adding to it information and values from additional areas. That
is, our strategy will be to begin with what is clearly feasible,
to build our collaborative patterns about an early success, and
then to expand in a systematic fashion to more ambitious goals.
We feel this is not only a good general management strategy but
the best way to build programming systems too.

Eventually. for instance it would be desirable for the system to
be able to learn from the data. First, however, the system must be
given the logic by which laboratory data are evaluated and understood.
We plan for development of the system in four phases.

Phase One: incorporate the medical logic which takes into
account the information which is available within the laboratory
itself: e.g. test results, quality control results, methodological
information.

Phase Two: Incorporate the additional medical logic which takes
into account information about the patient: first simple aspects such
as gender, age, race; then more complex concepts such as drug therapy,

operative status, clinical service assignment and provisional
diagnosis.

Phase Three: incorporate medical logic which includes concerns
for hospital function.

Phase Four: incorporate medical logic which attempts to link
to considerations which are outside the hospital setting.

Following is a more detailed description of the phased development.

Phase I. The aspect of the lab results which is of primary concern
within the laboratory hinges upon quality control considerations.
These are the first logical aspects which must be represented.

We are referring initially to thinking which currently goes
on strictly in the laboratory, previous to release of a test
result. Subsequently, there may or may not be significant
discussion between the laboratory director and the clinician
concerning further lab work and/or clinical concerns. Previous
to this stage, however, there is a great deal of evaluation done
now within the lab and based on laboratory on only partially
clinical grounds. Not enough evaluation of this sort is possible
with today's high volume instruments. This function can be greatly
enhanced by advanced computational techniques.

We would plan to introduce knowledge into the system
along the following lines:

1. Knowledge of the labs selected (likely we would start
    with hematology and clinical chemistry)

2. Knowledge of what tests are done, what methods are used,
    what parameters are estimated, what units are used. It
    should be noted that there are often multiple extant methods
    for a single determination, as well as multiple laboratory
    locations throughout the institution at which it might be

done. Methodology and unitage change continually. Since

a referral-type laboratory may do 3,000-5,000 different

determinations, it is a serious problem to choose a

representation which will be amenable to the endless updating

3. Knowledge of the kinds of patients and hospital locations.

4. Logic permitting an initial evaluation of the test result

for credibility. This naturally includes arithmetic

ranges, formats, etc.

5. Logic permitting evaluation taking into account other

results from examinations performed as a battery.

An example is the well known relationship between hemo-

globin and hematacrit.

6. Logic permitting evaluation of test result taking into

account laboratory quality control procedures and records.

We have recently completed an evaluation of the proposed

Bull statistic for control based on a weighted-moving-

average of mean corpuscular hemoglobin concentration,

which is a slight but still insufficient improvement on

the traditional method.

This is an example of the need to bring numerical methods

into alignment with the symbolic logic. In essence, this asks

the general question, is it likely the result is valid con-

sidering the quality of the particular "run" or batch

which produced the result?

The outcome of all the laboratory logic is the resolution

of the following questions:

a. Should the test be repeated using the same blood sample?

b. Is the issue important enough (or specimen identification

sufficiently questionable) that a new specimen must be obtaine

from the patient?

c. Should the result be reported to the clinician and

to the chart with some kind of qualification attached?

d. Is there a quality control problem in the laboratory

which requires immediate action?

e. Is there a breakdown in the clinical procedure

(ordering, specimen collection, etc.) which requires

immediate action?

Phase II. There are a number of clinical but relatively elementary

considerations which may be taken into account within the laboratory --

and which certainly should be taken into account by the knowledge-based

system we propose. Examples are:

1. Logic permitting evaluation of test results taking into

account basic information about the patient, i.e., age,

race, sex, and ward location.

2. Logic permitting evaluation of test results taking into

account previous test results in the same patient.

These pieces of information are often of critical

importance in evaluating the credibility or significance

of laboratory reports. Normal ranges, for example, vary

for some tests with age, race, and sex. Previous results

on a patient, to take another example, may be the first

clue to a mismarked specimen: the blood-from-the-wrong-

patient blunder which is so fundamental a problem for all

laboratories.

3. Logic permitting evaluation of test results taking into

account the general nature of the putative diagnosis (e.g.,

admitting diagnosis or treatment regimen).

It should be noted here that we are not proposing that
the system permit or encourage that clinical knowledge of the
patient influence the test result, but only the interpretation
of the result and the handling of the specimen. A general
diagnosis or even a treatment regimen can greatly influence
these matters. Plasma specimens from patients on oral anti-
coagulants, for example, usually should not yield normal
prothrombin times; indeed for these patients, "normal" is
abnormal and dangerous. The implication here is for interpre-
tation of the result, and when to report an "abnormality" through
the stat or emergency systems. Similarly, patients with
leukemias, especially under chemotherapy, often have remarkedly
elevated uric acids which have nothing to do with the usual
reasons for hyperuricacidemia.

The issues which are relevant at the patient or the clinician's
level hinge upon matters of test interpretation, the possibility
of needing to order further tests, the possibility of new diagnoses.
There is obviously an immense amount of logic which concerns
laboratory test interpretation in the context of all of the possible
clinical diagnoses and management problems. We are not proposing
to include this mountain of knowledge, which really pertains more
reasonably to programs such as Myer's INTERNIST System.

We propose to stop with knowledge which might reasonably be
construed to represent the conversation of the laboratory director
with the patient's clinical physician. It is difficult to specify
precisely this cut-off at the stage when we are only proposing
the system. The best indication of our intent might be provided
by an example.

77

It frequently happens that the lab director and a clinical hematologist will discuss a set of lab findings for a patient (with or without the question of errors in the findings) up to the point at which it is clear that the findings support the interpretation "iron deficiency anemia". This stage of reasoning represents a kind of intermediate between findings and diagnosis which AI systems sometimes call a concept. The semantic network system of Kulikowski, Amarel and Weiss, for instance, has such "concepts" within its logic. From the point of view of the logic we propose to write, this interpretation would be a proper termination, wholly supported by lab findings but requiring more clinical information about the patient than is obtainable from such paper systems as lab requisitions. The cause of the iron deficiency anemia would remain for another system to take up.

There are a host of such intermediate pathophysiological concepts which constitute a kind of proper frontier between clinical lab reasoning and more purely clinical reasoning. In practical terms, the resolution frequently is reached either by a telephone conversation between the lab director and the clinical physician, or by personal contact on such an occasion as rounds. We are not eager to automate the personal contact, although time does not permit enough of these discussions to occur; we would like to automate at least the decision to make the telephone call or appointment.

Most test results, even batteries of results do not permit an interpretation at the laboratory level. In some cases, we feel the logic could take us further. The most extreme case and the most complete logic we feel would end with a tentative pathophysiologic concept (such as anemia) and in selected important cases a decision on the part of the computer system to recommend the lab director call the clinician. Because of the limitations of

time, this is not a minor decision. Only the most important cases

should be selected for such conferences, whether telephone or in

person. A system with full and explicit logic should form a good

basis for such a decision. Furthermore, previous experience has

shown us that even our non-AI current lab monitoring systems

must bring together all pertinent (available) information about

a patient before bringing the abnormal report to the attention

of the user. This simple assembling of data aids current decision

making; we anticipate that assembly based on a more extensive logic

will prime a clinically useful discussion.

Phase III. Logic relevant to hospital function primarily concerns institutional

patterns. This includes changes in laboratory patterns, timeliness of

reporting, distribution of costs among services and patients, and

examination of interactions between procedures. For example, do

screening batteries including such tests as LDH's result in an inappropriate

number of repeat kinetic enzyme studies? These matters are derivative

measures of institutional function which are the natural by-products of

semantic understanding of the laboratory transactions. They would not be

examined until after the more fundamental logic in Steps I and II had

been dealt with.

Phase IV. Logic which links to considerations outside the hospital

environment.

It is difficult to detail these linkages ab initio. They are made

up   potentially of at least two separate concerns: derivation of facts

of general scientific interest; and the provision of linkages to educational

functions.

It must be emphasized that firm promises for such accomplishments

cannot be made. Still, one should point out some potentially important implications outside the immediate hospital realm, and should attempt to make the connections. A more or less modest scientific fact which could with luck result from the studies is the long awaited multivariate normal for application to multi-channel screening (Lezotte, 1977; Grams, 1977).

Building of instructional systems is beyond the scope of the present proposal, but provision of the connections is an inherent part of our plan. Good AI systems are (partly) characterized by their ability to defend their decisions. That is, a classification or advice provided from such an automated system can be challenged, and it can be expected the the system can recapitulate the rules or criteria which produced its conclusion. It is precisely this ability which should allow potential users outside the laboratory to benefit directly from the existence of such a knowledge-based system. We would hope to allow for this educational by-product usage by providing suitable means to challenge and converse with the system.

System building

We have given thought to the architecture of the proposed
system. It should be emphasized that this project is a long
term development in an area of fundamental importance to medicine:
namely, the knowledge which surrounds clinical laboratory testing.
We feel that there exists an adequate base of expertise in this
field at the University of Missouri, acknowledging of course that
we would utilize the full resources of the published literature and
that the knowledge and logic of the system would be subjected
to outside review by consultants as each major step was taken.
We do not, however, have an adequate experience in work in
artificial intelligence techniques per se to undertake the project
alone. It is clear that this competence exists in the group at
Stanford. We feel we have a sufficiently good working relationship
with Professor Feigenbaum and his colleagues that a joint develop-
ment will be successfully concluded.

The form of the actual computer representation has not been
selected. Our lab systems have used table driven assembly code for
years. The HCTC is collaborating with clinicians at UMC and computer
scientists at Rutgers to create a rule-based rheumatology consultant.
We wish to explore with Dr. Feigenbaum the possible appropriateness
of the imputational "blackboard" of the Hearsay system.

The knowledge-based system to incorporate clinical laboratory
expertise will be built on the SUMEX machine via the existing
time-sharing network. We have used terminal connections to SUMEX
for five years in connection with operation of the AIM network,
the SUMEX Executive Committee, and smaller experimental projects.
The communications are sufficient to support development of such

a system. At the same time, we recognize that it is inappropriate
(and probably impossible) for the SUMEX computer complex in
California to support a real-time service activity in Missouri.
Fortunately this is not necessary. Testing of the model in its
sequential versions against actual lab data in batches or bench-
mark sets can easily be done on a periodic basis. This will not
be a problem. Even the status of the quality control results can
be accessed and included in the model's operation in this fashion.
Since all transactions are recorded, one can accurately recreate
"real time" for any moment.

The issue of implementation of the full model in a real
laboratory setting is a separate problem. The system has not yet
been built, so we can't say what kind of computer would be needed
to run it. If we are correct in assuming, like other systems,
that a part of a PDP-10 is capable of running the model, then it
is not unreasonable to expect our laboratories to acquire this
level of computer support. The current lab systems are using a
combination of two PDP-12's, an IBM System 7, substantial
services of an IBM 370/158 (which is being replaced by an Amdahl
machine), and several microprocessors, including M6800's and LSI-11's.
All this does not add up to an AI machine, but we don't want it
to yet. There is a commitment to having computing gear at UMC,
and in most large clinical laboratories. At the same time, one
must acknowledge that the five year duration of the project
will doubtless see a continued reduction in the cost of computing
gear, as well as a continuation of the advances in hardware which
will have made AI techniques more realistic in the past. Machines
equivalent to DEC PDP-10's may well come to be offered for small amounts
of money in microforms. This kind of breakthrough is not necessary in order
for us to move over into an AI-based system. What is necessary is that

the system work well and be able to keep up with the changes in laboratory procedures which have plagued and almost destroyed previous systems. Our institution is currently supporting six full time programmers in a vain attempt to keep rigid old programming systems current with methodological and administrative changes. If the AI techniques succeed in producing a competent flexible software system, we feel that ongoing personnel savings will offset even large one-time hardware costs.

While the major model system is being built, we will naturally implement as improvements whatever parts of the logic are reasonable and feasible on the existing hardware. This is not difficult to imagine, because the current system is somewhat distributed already. It is through this means that we would expect to identify and hopefully to achieve cost savings and quality improvements. We assume that the major advances would come through implementation of the full new system. These should be calculated ahead of time. If the savings and improvements are "there", the project will have been successful and the system will be implemented as a whole at UMC and elsewhere.

Concepts to be included

There are certain general concepts which are suffused throughout
all elements of laboratory practice. These will necessarily be
incorporated in all phases of the proposed development.

These concepts include the following:

1. Statistical significance of testing, including
sensitivity - specificity of tests. This orientation
is inherent in lab work. Recent reports (Casscells, Schoenberger
Graboys, 1978; Ransohoff and Feinstein, 1978) indicate that it
is not well understood by the clinical users of laboratory services.

2. Related to this idea is the concept of normal, which
is very much dependent upon each particular laboratory,
and even upon specific methodologies. The knowledge of normal
ranges regarding the methodology and regarding age, sex, race,
and special circumstances of the test population must be
firmly associated in the system with each test specification.
The system must be able to defend its interpretations, and
hence to inform the user of the laboratory's assumptions
and adjustments to methodology.

3. The concept that automatic error detection is the
essential first step before interpretation of results is
attempted, and that the attempt at error detection must be
vigorous.   With the present systems we are able by careful
after-the-fact daily checking to recognize and correct errors in
data which have passed through the computer checks and have actually
been reported to the patient's chart. Two and one half percent of
results are in error. Of these 0.5% (in retrospect) actually represent

true technician or technologist methodological errors.

The remainder are a very mixed bag of clerical and

administrative errors. Our performance (which is probably

good compared with many wholly manual or semi-automated labs)

is the result of incorporating extensive computer editing

of the data. We long ago, for example, incorporated self-

check digit identification for patient and specimen numbers,

since we had shown that this category alone accounted for

half the errors detected by an earlier system (Lindberg,

Schroeder, Rowland, Saathoff, 1969).

Additional empirical methods of pattern recognition

have been developed for error deletion, and will be

incorporated in the proposed system. These include analysis

of electrolyte patterns, creatinine and others (Lindberg,

1968).

The current daily Abnormal Value Rounds in the laboratories

will provide an ideal work setting for the model development

and testing. Presently lab reports are transmitted by and

reviewed by the several computer systems. Special cases,

according to adaptive algorithms, are selected by the systems

for review daily by the chairman of the Department of Pathology,

Dr. Townsend, and his residents and staff. They currently

accept or reject the computer judgments based on their own

internalized judgments and upon additional data about the

patients which is obtained by going to see the patient and/or

the chart. It is this logic which should be represented in

the new programs.

4. Multi-step testing is a practice which has been common
to labs for decades. The logic is not always made explicit
to the user, and we feel there is an advantage in doing so.
The classic example is the serological test for syphilis.
Formerly, laboratories did a VDRL (for sensitivity), followed
in the positive cases by a Mazzini (for specificity). Currently
these have been replaced by the rapid plasma reagin test and
the fluorescent treponema antigen test. The same practice
is followed (appropriately) with many clinical enzyme tests
such as CPK and LDH, their kinetic counterparts and their
iso-enzyme extensions. Even more dramatic is the multi-step
or branching tree logic which is used by   coagulation
laboratories and thespecial immunology laboratories. The
questions to be addressed by the system include: what test
should be done first? What is available locally? What
subsequent test to do, dependent upon what initial results?
What statistical significance do the results have? What
further testing could be done? If this involves a remote
referral lab, how is the service obtained?

Essentially, this logic is quite subject matter dependent.
It is specific to the limited domains, but because of this,
also quite synonymous with expert behavior.

III.D.   Significance

The significance of a successful outcome would be:

1. Advances in basic knowledge representation techniques

2. Formal and public representation of a major field of medical expertise which will be of interest to all fields of medicine, health care, and information science.

3. Advances in techniques for remote collaboration on information system development. That is, we would be much further along on knowing how to share rare computational facilities and unique computer science competence with a broader, perhaps even national, medical community.

4. Improved understanding of evaluation of advanced health care technology.

The significance of a less than complete success would be lessened. Undoubtedly some of the representation and testing would be accomplished, since we will commence with the easiest part. If one's success were limited to this, the results would be of real importance but of interest primarily to laboratorians and computer scientists. These are an important part of the audience, but not the only ones we see for the complete system. The "downside risk", in other words, is minimal.

III.E.   Facilities available

The Health Care Technology Center can house the computer
component of the project at the University of Missouri-Columbia.
Space is available in a modern office building. The Center
provides library facilities, computer laboratory facilities,
telecommunication, etc. The Department of Pathology will be
providing access to the working laboratories as required. These
include Hematology, Chemistry, Microbiology, Clinical Microscopy,
Coagulation, Immunology and Anatomical Pathology services for
the University Hospital (440 beds), a similar arrangement for the
adjacent Harry S Truman Memorial Veterans Medical Center (480 beds),
the Mid-Missouri Mental Health Center (175 beds), and Rusk
Rehabilitation Center (100 beds). The combined laboratories
process 2,100,053 procedures a year.

Computer hardware per se includes 6 DEC LSI-11's; 3
M6800 systems; 2 DEC PDP-12's (tapes, disks, terminals);DEC PDP11/34;
IBM System 7; and multiple direct connections to the University
Network IBM 370/158 and 370/168 (both to be replaced by Amdahl gear).

The members of the Health Care Technology Center include
45 faculty from 14 University departments in 6 schools of the
Columbia campus.

The professional staff of the Department of Pathology includes
29   faculty and   20   residents and fellows. Only a subset
of the faculty are planned as active members of this project team,
but all are interested in the success of the venture and all are
available as needed for help on specific knowledge areas within their
own subspecialties.

III.F.   Collaborative arrangements

The system would be developed jointly with members of
Computer Science at Stanford and the Health Care Technology Center
at the University of Missouri-Columbia. Computer support for the
model system would be provided by the SUMEX computer facility. This
is an NIH supported national resource. Use of local computers at
UMC for data gathering, analysis, test implementation would be provided
free of charge. An exception is minor maintenance charges for
HCTC equipment. Telecommunications for approved projects are provided
by the SUMEX contract with TYMNET and ARPANET. Access to Net nodes is
provided by UMC WATS lines. In addition, the project would budget
funds to provide for frequent travel between the two schools.

Results of the project are to be published.

Stanford University is viewed as the primary submitter of the
proposed program project, with the University of Missouri-Columbia
supporting the application and taking responsibility for the Laboratory
Expert Project. Doctor Feigenbaum is the Principal Investigator for
the program project. Doctor Lindberg is viewed as Director of the
Laboratory Project.

PROJECT 3:   REFERENCES

1. Shortliffe, E.H., Axline, S. G., Buchanan, B.G., Merigan, T.C.
        and Cohen, S. N., "An Artificial Intelligence Program to
        Advise Physicians regarding Antimicrobiol Therapy".
        Computers and Biomedical Research, 6 (1973):1-17.

2. Weiss, S., Kulikowski, C. A., Safir, A. "Glaucoma Consultation by
        Computer". Computers in Biology and Medicine,8 (1978): 25-40.

3. Pauker, S. G., Gorry, G. A., Kassirer, J. P., Schwartz, W. B.
        "Towards the Simulation of Clinical Cognition: Taking a
        Present Illness by Computer". American Journal of Medicine,60,
        (June, 1976): 981-996.

4. Lawrence, S. V. "Internist: Computer Program Expressing Clinical
        Experience and Judgment of a Master Internist Constitutes a Unique
        Resource". Forum on Medicine (April 1978): 44-47.

5. Hicks, G.P. Evenson, M.A., Gieschen, M. M., Larson, F.C. "On Line
        Data Acquisition in the Clinical Laboratory". Computers in
        Biomedical Research Vol. III (Stacey and Waxman) New York:
        Academic Press, 1969, pp. 15-53.

6. Lindberg, D. A. B.: "Collection, Evaluation and Transmission of
        Hospital Laboratory Data". Proceedings 7th IBM Medical
        Symposium (1965): White Plains, New York, IBM, 1965.

7. O'Kane, K. C., Haluska, E. A. "Perspectives in Clinical Computing".
        In Advances in Computers, 16 (1977): Academic Press, 161.

8. Lezotte, D. C. "A Multivariate Laboratory Data Analysis System:
        Introduction". Journal of Medical Systems, 1, No. 3 (1977): 293-98.

9. Grams, R. R. "Progress Toward a Second Generation Laboratory Information
        System (LIS)". Journal of Medical Systems,(1) No. 3, (1977):263-74.

10. Casscells, W., Schoenberger, A., Graboys, T., "Interpretation by
        Physicians of Clinical Laboratory Results". New England
        Journal of Medicine 299, No.18 (November 1978): 999-1001.

11. Ransohoff, D. F., Feinstein A. R., "Problems of Spectrum and Bias in
        Evaluating the Efficacy of Diagnostic Tests". New England
        Journal of Medicine 299, No. 17 (October 26, 1978): 926-30.

12. Lindberg, D.A.B., Schroeder, J.J., Jr., Rowland, L.R., Saathoff, J.,
        "Experience with a Computer Laboratory Data System". In
        Strandjord, J. (ed), Multiple Laboratory Screening. Academic
        Press, New York, 1969, 245-55.

The undersigned agrees to accept responsibility for the
scientific and technical conduct of the research project
and for provision of required progress reports if a
grant is awarded as the result of this application.

_1/22/78_
Date

Donald P. Smith
Principal Investigator

IV.   CORE RESEARCH

IV.A.   Objectives of Research

The long term goal of artificial intelligence research at the Heuristic Programming Project (HPP) is to understand and build knowledge-based "intelligent agent" programs. Over the past decade we have studied such systems in the context of scientific and medical applications where human expertise for solving the problems was evident and where the difficulty of the problem seemed to lie just outside the boundaries of current AI methods. Because of the complexity of the applications, a significant part of the effort has been to make the expert knowledge of the problem explicit and to represent it appropriately in a knowledge base. This perspective has focussed attention on four areas for research:

(1) Representation — designing the symbolic structures for modeling the knowledge about a problem. Presently this phase is carried out by the system builders; we intend to codify the knowledge used to make such decisions, both as an aid to the system builders and ultimately to enable the programs themselves to choose appropriate representations.

(2) Reasoning — modeling the appropriate inference mechanisms for a problem and building systems that incorporate those models.

(3) Knowledge acquisition — designing systems that acquire knowledge by communication with human experts.

(4) Multiple uses of knowledge — designing systems that use the symbolic representation of the domain knowledge for additional purposes such as consensus building (accommodating conflicting advice from experts whose competence may be equal but whose "styles" vary), tutoring of human students by employing the knowledge base (both the information it contains and the way it is organized), and explanation (constructing a chain of rules which satisfactorily rationalize the system's behavior to an observer.

IV.B.   Background and Rationale

Artificial intelligence research at the Heuristic
Programming Project has utilized medical and scientific problems
to focus the research effort. For many different applications
over the last decade this has led to a cycle of research as
follows:

1. Form a collaboration with a scientist to work on a specific
    problem in a challenging and interesting area.

2. Propose a method for representing and manipulating the domain
    knowledge. This involves acquiring both formal and informal
    knowledge and developing a knowledge-based system that
    reasons with that knowledge.

3. Test the system. In this phase the method is pushed to its
    limits. The relationship between the design and the
    performance of the system is used as the basis for future
    development.

Both success and failure of a system can lead to further
research steps. When a system fails to solve a problem, the
seeds for further research can sometimes be found in the reasons
for failure. On the other hand, when a knowledge-based system is
successful, the desire to use it effectively uncovers a number of
additional needs. Thus, many of the topics of artificial
intelligence — such as the ability of a program to acquire
knowledge, or to explain its reasoning, or to manage updates in a
knowledge base — have grown out of programs that were at first
successful only at problem solving. From this experience has
come not only a set of approaches to building intelligent
systems, but also a broader understanding of what intelligent
systems should be like.

The following sections discuss the background information
about each of our major research areas. We will outline the
progress that has been made on this topic and identify the major
technological tools. Then in Section IV.C. we will discuss
our perception of the outstanding research issues and how we plan
to approach them.

IV.B.1.   Representation

93

One of the trends in our work has been to develop general purpose approaches for representing a broad range of knowledge in a knowledge base. This is illustrated by the Unit Package that has been developed for the MOLGEN project([40],[53]) for experiment planning in molecular genetics. In the figure below are two units from a MOLGEN knowledge base. The first unit represents the restriction-enzyme EcoRl; the second unit represents a problem-solving goal for an experiment.

```
NAME:                   ECOR1
SITE-TYPE:              STICKY-HEXA
3´-END:                 OH
5´-END:                 P
MODE:                   NON-PRECESSIVE
MOLWT:                  28500
SUBSTRATE:              DNA
RECOGNITION-SITE:

                        1  2  3  4  5  6  7  8

                        G    A  A  T  T     C
                        _____


                        _____

                        C     T  T  A  A    G
                        16 15 14 13 12 11 10 9
```

```
NAME:                   LAB-GOAL-1
STATE:                  A CULTURE with
                           ORGANISMS = A BACTERIUM with
                                          EXOSOMES = A VECTOR with
                                             GENES = RAT-INSULIN
CONDS:                  (PURE? ORGANISMS CULTURE)
```

The usual way of using the Unit Package is to define general knowledge before specific knowledge. For example, general knowledge about enzyme, nuclease, and restriction enzymes would be entered before the specific knowledge about a particular restriction enzyme like EcoRl. The Unit Package is designed to encourage the use of description, such as the description of a culture in the second unit above. These descriptions are used for checking new information as it is entered and for pattern-matching operations that are part of a reasoning step. Reference [52] describes the Unit Package and compares it to other work on representation.

The examples above have illustrated the representation of "object-centered" or "noun-like" knowledge. Every reasoning program also contains a representation of the inferential

knowledge. In the first version of the DENDRAL program, this kind of knowledge was represented as a program. This choice of representation had the consequence that a chemist could not enter new knowledge into the program (because he could not be presumed to be an expert programmer). Also, since the program structures were not understandable by the program itself, facilities for explanation of DENDRAL's reasoning had to be built into each part of the program. In the MYCIN program [51], developed more recently, the inferential knowledge was moved out of the program and into a knowledge base represented as production rules. This representation, because it was closer to the experts' representation than DENDRAL code was, allowed us to develop programs that could acquire rules from physicians. It also allowed the system to generate its own explanations by examining the rules it had used. Production rules illustrate many of the themes which run through our work on representation.

(1) Explicitness — Knowledge is encoded in a knowledge base and not just in programs. (For example, production rules are used to make inferential knowledge explicit.) The distinction between knowledge being in a program or in a knowledge base is a crucial one, for our purposes. Information encoded as a program can be run, and initially coded, more easily and quickly. However, as the program grows, it becomes more and more difficult to add new knowledge : its relationships to all the other knowledge must be considered and programmed explicitly. The latter method, storing knowledge in a separate data structure, a "knowledge base", enables the pieces of knowledge to be accessed and manipulated just like data. While their use, their running, may be somewhat slower, the system builder can now enter data in modular fashion, without much concern for the rest of the items in the knowledge base. He can give the system the knowledge it needs to reason about its own knowledge base.

(2) Modularity — Knowledge is encoded in independent "chunks" as far as possible. (Production rules can be added or deleted from a knowledge base to change its problem-solving behavior.) The concepts chosen to represent the chunks of knowledge are those which are natural and useful to a domain expert. This is useful both if the expert is to input rules directly, and if he is to be convinced by the system's explanation of its behavior.

(3) Uniformity — Knowledge is represented so that it can be manipulated by general purpose programs. (Production rules and frames are two of the uniform methods for which we have general purpose processing routines.)

Our perception of the outstanding research issues in representation is discussed in Section IV.C.1.. As can be seen from the examples above, how knowledge is to be used is important in determining how it should be represented. With more uses for knowledge — explanation, tutoring, problem-solving — come more constraints on its representation.

IV.B.2.   Reasoning

The first step in creating a problem-solving system is to develop and test a method for reasoning. In the DENDRAL program[11] for inferring chemical structures from mass spectrometry data, the reasoning framework that we tested was called the Generate-and-test paradigm. This consisted of (1) an exhaustive generator of all possible solutions (chemical structures) and (2) a set of pruning rules which used the mass spectrometry data to eliminate inconsistent answers. One of the issues that became relevant in studying this reasoning framework is the combination of possibly contradictory evidence. Data in many problems is incomplete and errorful; there is seldom a perfect match between an internal model and empirical data. Even if DENDRAL had a perfect model of how mass spectrometry data corresponds to chemical structures, the data from any particular run of a mass spectrometer are erroneous with respect to both extraneous and missing data. In DENDRAL, an overall domain-specific matching function was used which reflected a priori probabilities of errors in the data. Recently we have reexamined this problem in the context of the GA1 program[53] which solves an analogous problem from molecular genetics.

For the MYCIN program we used backwards-chaining as a reasoning framework. This method develops a line of reasoning by chaining together MYCIN's inference rules (production rules) backwards from the goal of making the diagnosis towards the available evidence. This particular reasoning framework has proved especially convenient for developing computer explanations of the program's reasoning. To deal with imperfect evidence and inexact rules of inference, a mathematical model of certainty based on numeric "certainty factors" was developed. This constitutes a model of "plausible reasoning". In order to test the MYCIN approach in other domains, a domain independent package, EMYCIN (for "Essential MYCIN") has been created and is being utilized in other applications discussed elsewhere in this proposal.

When MYCIN is chaining back through its inference rules and discovers a need for information that cannot be inferred, it stops and asks for it. This approach is appropriate only when

96

there is a way of supplying data as needed by the reasoning program. For some applications, such as signal interpretation, it is better for the program to make use of whatever it knows, because there is little chance that specific items of information can be supplied on demand. Further limitations of a simple backwards-chaining model are (1) it is unidirectional, hence cannot mix top-down and bottom-up processing and (2) it is exhaustive, hence less efficient than approaches that reason hierarchically by working with abstractions.

An alternative reasoning model which does not have these limitations is the "cooperating knowledge sources" model developed for the HEARSAYII [20] system and incorporated in our AGE-I program. This model consists of (1) the "blackboard", a global data structure which holds the system's hypotheses, and (2) a set of "knowledge sources" (KSs) which contain the inference rules for the system. Because of gaps in the theory and implementation of the individual KSs and noise in the data, the KSs are individually incomplete and errorful. A version of the "hypothesize and test" paradigm is used which emphasizes cooperation (to help overcome incompleteness in both knowledge and data) and cross-checking (to help correct errors). During the hypothesize part of the cycle, a KS can add a hypothesis to the blackboard; during the test part of the cycle, a KS can change the rating of a hypothesis in the blackboard. This process terminates when a consistent hypothesis is generated satisfying the requirements of the overall solution or when knowledge is exhausted. The power of the blackboard — over, say, a uniform QA4 assertional net — is its structure: it is n-dimensional, where the dimensions have some meaning (time, level of abstractness, geographic location, etc.). Hence each rule can know what part(s) of the blackboard to monitor, and each hypothesis is carefully placed at a meaningful spot on the blackboard. This is a simple but powerful type of analogic modelling of the domain.

Two research programs based on this paradigm have been developed by our group [43]. One is the CRYSALIS program for interpreting x-ray crystallography data and the other is a military signal interpretation program. In these programs the HEARSAY model was extended by (1) extending the blackboard to allow for several independent hierarchical relationships among data and hypotheses and (2) extending the control structure.

In each of the examples above, our study of reasoning methods always starts in the context of a problem in a scientific or medical domain. We then generalize the method and package it for further testing in other domains. When a framework for reasoning works well enough, research on other artificial intelligence topics, such as explanation or knowledge

acquisition, often follows. Our perception of open research issues in reasoning methods is discussed in Section IV.C.2..

IV.B.3.   Knowledge Acquisition and Management

One characteristic of the domain problems we have studied is their requirement for a substantial amount of domain expertise. Goldstein addressed this point in [26]:

> Today there has been a shift in paradigm. The fundamental problem of understanding intelligence is not the identification of a few powerful techniques, but rather the question of how to represent large amounts of knowledge in a fashion that permits their effective use and interaction. This shift is based on a decade of experience with programs that relied on uniform search or logistic techniques that proved to be hopelessly inefficient when faced with complex problems in large knowledge spaces.

The relevant problem solving knowledge includes much formal and informal expertise of the domain expert; it also includes many mundane facts and figures that make up the elementary knowledge of the domain. Before a computer system can solve problems in the domain, this information must be transferred from the expert to the computer.

Over the last decade, there has been some encouraging progress along this dimension. In DENDRAL, the rules of inference about mass spectrometry had to be put in machine form, but knowledge acquisition by the program from the chemist was beyond our technology. Knowledge was added by a painstaking process in which a computer scientist together with a chemist learned each other's terminology and then wrote down the chemical rules for the simplest kinds of chemical compounds. Then the computer scientist entered the rules into the computer and tested them and reported the results back to the chemist. The reward for this effort over several years was a program with expert-level performance.

It is interesting to compare the knowledge acquisition effort of the DENDRAL program with that of a more recent program

— PUFF, the system for diagnosing pulmonary function disorder. In contrast with DENDRAL, PUFF was created in less than 50 hours of interaction with experts at PMC and with less than 10 man-weeks of effort by the knowledge engineers. Part of this tremendous difference in development time is due to the fact that the domain of pulmonary function is much simpler than mass spectrometry. However, the main reason that the development was so rapid is that PUFF was built with the aid of an interactive knowledge engineering tool, EMYCIN. When knowledge engineers at the Heuristic Programming Project started the PUFF project, they already had a reasoning framework in which to fit the problem and an "English-like" language for expressing the diagnostic rules. The facilities that make EMYCIN such a powerful tool are the direct result of the core research over the last five years on the MYCIN program.

Another dimension of progress closely related to knowledge acquisition is knowledge management, that is, management of the global structure of a knowledge base. A knowledge base is more than a set of isolated facts: its elements are related to one another. In the DENDRAL program, all of the knowledge was represented as programs and LISP data structures. If changing one part of the program meant that another part had to be changed as well, the programmer had to know that. As programs or knowledge bases get large, this kind of effort becomes substantial. A system becomes too large to maintain when no one can remember all of the interactions and every change introduces bugs. TEIRESIAS[15] extends the idea (developed initially in automatic programming research) that a system can aid substantially in identifying sources of errors and can take on some of the responsibility for making changes.

Research issues in knowledge acquisition and management are discussed in Section IV.C.3..

IV.C.   Methods of Procedure

We are interested in exploring the effects of new ideas about knowledge based programming on a variety of systems to effectively test the generality of these ideas. Each of the topics in the core research area will be developed in the context of more than one example program (see discussions of Projects 1-3).

The expert systems developed at the Heuristic Programming Project over the last decade can be used as tools for the

development of the core research topics. Each of the biomedical domains has particular aspects that can be utilized in this work: the MOLGEN program for molecular genetics research has methods for representing experiment planning, the MYCIN program for infection disease diagnosis and therapy has a well developed rule set, the PUFF program for pulmonary function test interpretation has a small rule set, and the VM program for interpreting physiological measurements from the Intensive Care Unit has a knowledge base that emphasizes knowledge that changes over time.

IV.C.1.   Representation

In Section IV.B.1. we traced our work from specialized representations as in the DENDRAL program to representations of more general applicability — such as our production rule and frame methodology. Today's representation systems, even the "general" ones, do not solve all of the problems that we are encountering in our research. In most science, methods which are general are also weak. There seems always to be a need to tailor aspects of a representation to particular problems. The following representation issues stand out in our work:

Time-based knowledge

Several problems which we are working on involve situations that evolve over time. In the Ventilator Management (VM) program [21], time enters as instrument data that varies over time. The program must correctly track the stages of treatment on the treatment machines. In the RX program [5] for reasoning from time-based clinical data bases, statements about disease and treatment of patients need to be adequately quantified over time. In the MYCIN [51] work, we want the system to be able to resume a consultation session about a patient and appropriately update new knowledge about the patient as treatment progresses. In the MOLGEN project [40], the experiment planning program must plan a sequence of steps. It must predict how the laboratory objects will be changed over time as the manipulations proceed. The basic issues common to these projects are (1) time-specified reference to objects and (2) tracking causal changes on objects over time. While these problems do not seem conceptually difficult, they do require extensions to the representational tools which we have available.

Grain Size in Complex Systems

Among the virtues of production rules [6] are (1) their modularity allows easy addition and modification of inferential knowledge and (2) they can be written in such a way that their grain size seems appropriate for explanation systems. As we move toward hierarchical reasoning methods the grain size of individual production rules seems too small for coherent explanations. Just as the reasoning methods work with abstractions to reduce the combinatorics, explanations of this should also be abstract.

At present, the problem of factoring knowledge is an opaque art. When a frame-structured representation is used, a knowledge engineer makes decisions about what facts to group together. This decision takes into account indexing during problem solving and the interactions among items in the knowledge base. In hierarchical reasoning methods knowledge is viewed with a varying grain size; it starts with an abstract conceptualization at the beginning of problem solving and moves toward finer detail as the solution proceeds. Although we have some understanding of how to organize a body of knowledge hierarchically, much work remains to be done to make the best use of that organization during knowledge acquisition and problem solving.

Matching representation methods to problems

In our current systems, a knowledge engineer must learn the particulars about a problem and then pick or develop an appropriate representation. We would like to extend current AI ideas in the design of a system which takes more responsibility for choice of representation. Such a system will select or modify its representations combining the knowledge of the limits and advantages of representations with the knowledge of its own needs.

IV.C.2.   Reasoning

In Section IV.B.2.} we traced our research on methods of reasoning from the Generate-and-Test paradigm (DENDRAL, GA1), to backwards chaining (MYCIN, EMYCIN, PUFF), to the cooperative knowledge sources model (CRYSALIS, HASP, AGE-1). In this section we discuss core issues related to these reasoning models as well as some ideas for new models.

Incomplete Reasoning

---

[6]See [16] for a discussion of different ways of using this formalism.

One of the themes in all of our methods of reasoning is the treatment of inexact and incomplete knowledge. One of the difficulties which we have perceived in MYCIN's simple CF model is that the representation is inadequate for discriminating between (1) absence of evidence and (2) evidence of absence. This example illustrates how the needs of the reasoning program have to influence the fundamental representations used in the system.

Reasoning with Abstractions

The availability of the Unit Package [52] has broadened our capabilities for representing abstractions. For example, an organism can be variously described as "a bacterium", "E.coli K-12", "a bacterium that is grampositive", or even "a bacterium with a vector which has the rat-insulin gene". A reasoning program can use the descriptions available in the Unit Package as abstractions in its reasoning process. We are currently using this idea in the MOLGEN project for reasoning·· about experiment planning.

Orthogonal Planning

One of the themes in our representation work is to make knowledge explicit for general processing. We have carried this theme into an experimental framework for reasoning being developed currently in the MOLGEN project. The idea is to make the reasoning operations, which are carried out by a planner, explicit in the knowledge base. These operators then implicitly define an abstract "planning space". Our hope is that this will provide a computer with a planning method more powerful and flexible than previous hierarchical planning methods. The feasibility of this approach is currently being tested.

Matching Reasoning Methods to Problems

One of our long term goals in developing and understanding reasoning methods is to develop a theory for matching reasoning methods to problems. Such a program would combine knowledge of the limitations of available reasoning frameworks with the needs of an application to aid in the design of a knowledge based system. We have started on this problem with the research of the AGE project within the HPP.

IV.C.3.   Knowledge Acquisition and Management

In Section IV.B.3., we traced our work on knowledge acquisition from the DENDRAL program, where knowledge was acquired by a knowledge engineer and then programmed into the system, to the PUFF example where the EMYCIN package greatly accelerated the creation of a consultation system for pulmonary function diagnosis.

Three Phases of Knowledge Acquisition

As a result of our recent experiences with the SACON program [3], we have found it useful to characterize the knowledge acquisition process as occurring in three distinct phases. We have done the most research on the third phase and plan to work our way towards the first phase.

(1)   Framework Identification. The first phase corresponds to making initial decisions about the typical advice the consultant will give and the major reasoning steps the consultant will use.

(2)   Acquisition of Fundamental Concepts. This is followed by an extended period of defining parameters and objects. These objects form the fundamental vocabulary of the domain. Using this initial domain vocabulary, a substantial portion of the rule base is developed. This process, captures enough domain expertise to allow the consultation system to give advice on the large number of common cases.

(3)   Acquisition in a Well-Developed Knowledge Base. In the final phase, further interactions with the expert tend to refine and adjust the established rule base, primarily to handle more obscure or complicated cases. In this phase, the system can draw on examples from the knowledge base to guide the acquisition process.

Previous work on the TEIRESIAS program [15], which explored one possible method for handling the "final phase", will provide the basis for our research in knowledge acquisition. This phase of the acquisition task utilizes the large body of knowledge to set the appropriate context for understanding new facts.

Consistency

Developing an understanding of the automatic management of knowledge during and after its acquisition is an important aspect of our research aims. The knowledge base consists of the totality of concepts and relations between concepts that have been presented to the program. We will investigate methods for determining the consistency of the aggregate knowledge base.

The quality of the knowledge base is improved through experimentation. Cases are run (for medical domains) by selecting a diverse set of patients and comparing the results to the conclusions of our expert. When the results don't match, the knowledge base must be updated to account for those changes. Two operations are important for this process: (1) the ability to determine the piece or pieces of knowledge that must be changed and (2) determining that changing the knowledge to correct the results on one patient will not produce incorrect results when applied to another patient.

Another possibility is to identify and, in effect, live with inconsistency, just as people apparently do. Predominantly rational behavior may be evinced by a system which does not satisfy consistency requirements. The key test is whether the elimination of any "inconsistent" rule makes the system behave better or worse in the long run. This is closely tied to consensus-formation, as discussed in the next section.

IV.C.4.   Multiple Uses of a Knowledge Base

We are exploring many additional uses of the knowledge base beyond the performance aspects for which we acquired the knowledge. Three areas are of interest: using the knowledge for explanation of the reasoning steps of the program, using the knowledge for intelligent teaching about the domain, and using the knowledge base as a vehicle for building consensus among experts.

Explanation

The use of explicit inference rules in a knowledge base has made it possible to generate an explanation of the programs' reasoning steps. While this has been achieved in the "backwards chaining" reasoning model, it is more difficult in the reasoning methods which reason hierarchically. We will examine methods for modifying the level of explanation based on the abstractions used by the program and a model of the user.

Tutoring

The act of explaining the knowledge has led to the problem of using the knowledge base for tutoring purposes. Our initial experiment with this in the MYCIN framework [12] demonstrates the potential educational value of this use of the knowledge base. Under another proposal (pending to ONR & ARPA) we will be exploring strategies for presenting the contents of a knowledge base represented as a set of rules. Here we propose to extend those methods for relating to the user the contents of knowledge bases stored in other representations.

Consensus Building

We propose to investigate approaches for building consensus among experts. Because the strength of consultation programs will in large part lie with their ability to pool knowledge from several sources, it is important to recognize apparent differences of opinion among experts and to assist, when possible, with arriving at a consensus. This represents another version of the consistency checking problem: comparing the ramifications of multiple versions of knowledge and providing the capability to guide an interaction in which such differences are "ironed out". Of course there may be times when both versions of the knowledge may need to be stored and appropriately flagged so that users can select which experts' opinion they will follow during a consultation. The experts may wish to select a style of reasoning (e.g., empirical vs theoretical), rather than a particular individual's set of rules. Ultimately, the system itself may be able to choose from differing advice in its knowledge base.

All of these areas require some augmentation to the knowledge base to provide the causal reasoning steps upon which the knowledge is tied. This allows a program to explain why a particular rule was written in addition to telling how the rule was used to make a particular conclusion. Similar needs have been shown in the use of a rule base for tutoring and for determining consensus among experts [37]. Often, a rule will be put into the system cast in a much more specific form than that to which the knowledge truly applies. One task to investigate is how to generalize to just the proper level. More complex still are the subtle changes that accompany a rule as it is generalized (e.g., changing certainty factors).

IV.D.   Significance

The significance of this work is twofold:

1. Understanding how to represent inexact and incomplete
    knowledge symbolically so that a system can perform complex
    intelligent processes — like diagnosis and explanation.
    This work expands the boundaries of what we understand how to
    do with computers.

2. Investigating the fundamental questions that underlay the
    development of domain-independent tools of AI discussed
    elsewhere in this proposal.

        One of our ultimate goals is to understand the techniques
employed in building such programs. It has always been difficult
to determine if a particular problem-solving method used in a
particular knowledge-based program is domain-specific or whether
it can generalize easily to other domains. In current knowledge-
based programs, the domain knowledge and the manipulation of it
using AI techniques are often so intertwined that it is difficult
to uncouple them, to make a program useful for another domain.
This long range goal, then, is to isolate AI techniques that are
general, to determine the conditions for their use; to build up a
knowledge base about AI techniques themselves. We will carry out
our research with this question in mind: what are the criteria
determining whether a particular problem-solving framework and
representation system is suitable for a particular application?

V.   FACILITIES AVAILABLE

V.A.   Hardware

All computing work will be carried out initially on the
SUMEX facility, a dual processor DEC KI-10 system running TENEX.
The system is located at Stanford, but is supported by NIH under
grant RR-0785 as a national resource for the study of
applications of artificial intelligence to problems in biology
and medicine. It has available a wide variety of advanced
programming languages (e.g., INTERLISP, SAIL), and support
programs (e.g., text editors), as well as powerful file handling
and storage management capabilities. Resources available at no
cost to this program include CPU usage and disk storage, while
access is via local dial-up lines and three networks (TYMNET,
TELENET, and ARPANET).

Within the next 18 months the SUMEX installation is also
scheduled to receive a PDP-20/20 system that will be interfaced
with the currently existing PDP-10. The new machine is intended
for service-related applications of artificial intelligence to
medicine, and some of our programs, once operational, would most
appropriately be run on this machine. The machine will be used
by other projects, however, and may occasionally be scheduled for
sole use by one of these. Thus SUMEX can make no commitment to
provide scheduled service to medical personnel wishing to use the
programs routinely. The PDP-20/20 hence will function as a
prototype for the kind of dedicated small machine that may
eventually operate in the clinic.

V.B.   Software and Personnel

Our proposal is to build on the knowledge representation
and control techniques developed during work on the MYCIN,
Molgen, PUFF, and AGE systems in the Heuristic Programming
Project. New programs and data structures will, of course, be
required. Starting with existing software packages, however, is
a considerable advantage over developing the software - and
design experience - de novo. The base language will continue to
be INTERLISP.

In addition to the computing power and the large collection
of existing software, access to the SUMEX system also offers the

benefit of being a part of the SUMEX-AIM community.   The SUMEX
user community includes a wide range of researchers in artificial
intelligence united by a number of common interests. We have
found our interchanges with them in the past to be very useful,
and expect this to continue.

VI.   COLLABORATIVE ARRANGEMENTS

Formal collaboration with Dr. Lindberg's group at the University of Missouri is the natural result of many years of informal exchange. The formal arrangement between the two institutions is that Dr. Lindberg's project will be funded as a subcontract from Stanford, with budget as indicated in the budget section.

There is a long history of successful collaboration between the Stanford Medical School and the Computer Science Department. The SUMEX Computer Facility is a physical demonstration of this collaboration, while the large number of interdisciplinary research publications is more evidence. In part, this is due to the physical proximity of the two groups; but more importantly, it is due to common interests and common goals. The SUMEX facility itself has removed many of the communication barriers which often halt interdisciplinary research.

VII.   <u>PRINCIPAL INVESTIGATOR ASSURANCE</u>

    The undersigned agrees to accept responsibility for the
scientific and technical conduct of the research project and for
provision of required progress reports if a grant is awarded as
the result of this application.

Jan. 30, 1979   _Edward A. Feigenbaum_
Date   Principal Investigator

VIII.   APPENDICES

VIII.A.   APPENDIX A -- Annotated MYCIN Typescript

In the following pages we have included many detailed examples of the MYCIN program in operation. These exemplify both the accomplishments and the limitations of the work we have done so far. Although we are not proposing expansion of the program's infectious disease knowledge at this time, these examples should help illustrate the kinds of capabilities that we intend to develop in a system for oncology protocol management.

The examples in this appendix include the following:

Section I - A sample production rule, translated into English.

Section II - Instructions printed for new users if they request assistance when trying MYCIN for the first time.

Section III - Free-text case summary that may be entered by a physician for purposes of case identification in the future.

Section IV - Detailed example of a consultation session for a patient with meningitis; the WHY and HOW commands of the reasoning-status checker (RSC) are also demonstrated.

Section V - Interactive session with the general question answerer (GQA) regarding the consultation session in Section IV.

Section VI - Example of MYCIN's ability to assist with antibiotic dosage modification in renal failure patients; note that the program can also explain its decisions at this specialized task.

Section VII - Example of a graphical option we have developed which permits interested physicians to display a chart estimating the steady state blood levels of an antibiotic at a variety of regimens for modified dose or dosing interval.

Section VIII - Example of a subsystem of MYCIN in which the user can circumvent much of the extensive consultation session demonstrated in Section IV. If a physician is relatively certain of the infection and organisms to be treated, he may specify these as shown and MYCIN will simply assist with therapy selection.

Section IX - Example of MYCIN's ability to rerun previously stored patients <u>and</u> to interact with an expert when a problem in performance is identified. Note that MYCIN and the expert have a "discussion" in which a missing rule is identified. The physician tells MYCIN the missing rule (in English) and the program translates it into its internal LISP representation. The case is then run again to see if the performance improves with the new rule in place.

I. Sample Rule with Addditional Stored Information

RULE300

[This rule applies to all cultures and suspected infections, and is tried in
     order to find out about the organisms (other than those seen on cultures
     or smears) which might be causing the infection]

   If: 1) The infection which requires therapy is meningitis, and
        2) The patient does have evidence of serious skin or soft tissue
           infection, and
        3) Organisms were not seen on the stain of the culture, and
        4) The type of the infection is bacterial
   Then: There is evidence that the organisms (other than those seen on
          cultures or smears) which might be causing the infection is
          staphylococcus-coag-pos (.75) streptococcus-group-a (.5)

Author: YU

Literature: G. Karalazin "Sickle-cell anemia – CLINICAL MANIFESTATIONS
     IN 100 PATIENTS" AmJMedSci 269:51 1975.
        E. Barret-Connor "Acute pulmpnary disease and sickle-cell anemia"
     ARRD 104:159 Aug 1971.
        M. Robinson "Pneumococcal meningitis in sickle-cell anemia"NEJM
     274:1006 1966.

II.   Instructions for Users at the Start of
      Infectious Disease Consultation (optional)

MYCIN 3-Jun-78 ...

Special options (type ? for help):
**
Instructions? (Y or N)
** YES

     This is a computer program named MYCIN that has been designed to
advise you regarding an appropriate choice of infectious disease
therapy.   It is understood that you have a patient who may have an
infection.   Please answer the following questions, terminating each
response with RETURN. To correct typing errors, use the DELETE key to
delete single characters, <ctrl>W to delete a word, and <ctrl>Q to
delete the whole line.

     If you are not certain of your answer, you may modify the response
by inserting a certainty factor (a number from 1 to 10) in
parentheses after your response. Absolute certainty (10) is assumed
for every unmodified answer. It is likely that some of the following
questions can not be answered with certainty.

     You may change an answer to a previous question in two ways.
If the program is waiting for a response from you (that is, has typed
"**"), enter CHANGE followed by the number(s) of the question(s)
whose answers will be altered.   You may also change a previous
answer at any time (even when the program is not waiting for a
response from you) by typing <ctrl>F (Fix), which will cause the
program to interrupt its computation and ask what you want to change.
(If the response to <ctrl>F is not immediate, try typing the RETURN
key in addition.) Try to avoid going back because the process
requires reconsidering the patient from the beginning and therefore
may be slow.

     Note that you may also enter UNK (for UNKown) if you do not know
the answer to a question, ? if you wish to see a more precise
definition of the question or some examples of recognized responses,
?? if you want to see all recognized responses, the word RULE if
you would like to see the decision rule which has generated the
question being asked, the word WHY if you would like to see a more
detailed explanation of the question, or the letters QA if you would
like to interrupt the consultation in order to ask questions
regarding the decisions made so far in the consultation. If you are
ever puzzled about what options are available to you during a
consultation, enter the word HELP and a list of options will be
listed for you.

     Sample Response [user input follows the "**"]
        Does the patient have a risk factor for tuberculosis?
        **?
        One or more of the following are considered risk factors for tb:
        a) positive PPD (5TU), b) history of close contact with a

114

person having active tb, c) household member with a past
history of active tb, d) chest X-ray showing apical scarring,
e) granulomas seen on biopsy of any organ tissue.

expected responses are: YES NO
Enter HELP for user options.
** YES
SUMMARY:
  (type ctrl-O to abort printout)
  UNK - answer not known
  ?   - Rephrases the question and gives examples of recognized
         responses
  ??   - prints a list of all recognized responses
  RULE - prints the current decision rule
  QA   - program enters question-answering mode
  CHANGE   - go back and re-request answer to question number
  COMMENT - enter comments about any aspect of the system's
         performance. Your comments will be forwarded to those in
         charge of the MYCIN program.
  WHY - gives high-level explanation of the current reasoning
         chain that provoked this question.
  HOW   - explains HOW the system will achieve a goal referred to
         by number   in a previous explanation.
  EXPLAIN - provides a more detailed explanation of a previous
         answer given by a WHY command.
  FORGET - resets the explanation of the reasoning chain back to the
         lowest level, as if you never used the WHY/EXPLAIN commands.
  STOP - Halts the program without completing the consultation.
  SAVE - Halts the program, saving the current patient on a disk
         file, retrievable at a later date.
  HELP - Prints this list

III. User's Description of Patient
This will be patient 538
If you would care to enter a brief summary of PATIENT-538's case, to be used
for easy future reference to this file, please do so now (type as many lines
as required, terminate with a control-Z):

** 34 yr old white male with severe burn. After 15 days
** of hospitalization pt became disoriented and febrile. On 16th
** hospital day LP performed. PMN 100, PMN%99, Pro 280, Glu 5 (periph
** blood glu unk). WBC 1900 with 88% PMN and 5% bands. Pt had drug-
** induced aplastic anemia. Current med at time of LP: Cephalothin.
** Final org: Pseudomoas-aeruginosa. Correct Rx: Ideal would have been
** Carb + Tobra (or Genta). Same as pt 504 except gram stain of CSF
** omitted.
   Z
**

IV.   Infectious Disease Consultation Session
         [User's responses follow double asterisks]

         --------PATIENT-538--------
   1) Patient's name:
   ** PT538
   2) Age:
   ** 34 YEARS
   3) Sex:
   ** MAEL
=MALE
   4) Race:
   ** WHY

[i.e. WHY is it important to determine the race of PATIENT-538?]

[1.0] The race of a patient is one of the 5 standard parameters that identify
     a patient.
      This may be useful later in the consultation. For example, it is
     relevant in determining the organisms (other than those seen on cultures
     or smears) which might be causing the infection.

     [back to question 4...]
     ** CAUCASIAN

Please supply one answer under each heading. If your response is not
understood, you will be prompted by the standard questioning method.
(enter ? for normal prompting, * for questions that are not applicable).

Please enter the results of positive cultures using the following table.

site	culture number	date collected	date| positive	method| of collection [urine,sputum only]

   5)**DONE

Please enter the results of pending cultures in the following table.

site	culture number	date collected	stain| examined	method of| collection [urine,sputum only]
   6)** | CSF | 1234 | 30-Jan-77 | YES | * |

         stain of morph|
         smear
         [type DONE if no organism was seen on the smear]
     7)**DONE

site	cul	date coll.	smear examined?|	coll. method|
   8)**DONE

Please enter negative cultures in the following table.

117

```
     site          culture   date|          method of|
                   number    collected      collection
                                            [urine,sputum only]
9)**DONE
```
   10) Do you suspect Pt538 may have an infection at a site from which you
       have not obtained culture specimens?
   ** NO

Please enter information on current antimicrobials.

```
     drug                    route of           date|
     name                    administration     started
11)**   CEPHALOTHIN          IV                 23-Jan-77
12)**DONE
```

Please enter information on prior antimicrobials.

```
     drug          route of          date          date|
     name          administration    started       stopped
13)**DONE
```

[Considering organisms (other than those isolated from cultures) which might
be present...]

   14) Has Pt538 recently had symptoms of persistent headache or other
       abnormal neurologic symptoms (dizziness, lethargy, etc.)?
   ** YES
   15) Has Pt538 recently had objective evidence of abnormal neurologic signs
       (nuchal ridigity, coma, seizures, etc.) documented by physician
       observation or examination?
   ** YES

The CSF cultures will be considered to be associated with MENINGITIS.

   16) Please give the date on which clinical evidence (symptoms, signs, or
       laboratory tests) of the meningitis first appeared.
   ** 29-Jan-77 22:15
   17) Does Pt538 have an abnormal chest X-ray?
   ** NO
   18) Does Pt538 have one or more of the following risk factors for tb: a)
       positive intermediate strength PPD, b) history of close contact with a
       person with active tb, c) a household member with a past history of
       active tb, d) apical scarring on chest xray, e) history of granulomas
       on biopsy of liver, lymph node, or other organs.?
   ** NO
   19) Does Pt538 have any cutaneous lesions or rash on physical examination?
   ** NO
   20) Does Pt538 have evidence of ocular nerve dysfunction?
   ** NO
   21) Is Pt538 a compromised host (e.g. alcoholic, sickle-cell-disease,
       immunosuppressed)?
   ** YES
   22) Is Pt538 immunosuppressed (e.g. corticosteroid therapy, cytotoxic drug

           therapy, radiation therapy, leukemia, lymphoma)?
     ** NO
     23) For how many days has Pt538 had abnormal neurologic signs?
     ** 7.2 HOURS
     24) For how many days has Pt538 had neurologic symptoms?
     ** 1 DAYS
     25) Has Pt538 had recent evidence of tender or enlarged salivary glands?
     ** NO
     26) Has Pt538 been exposed to any contagious disease recently (e.g.
         meningococcal disease, mumps) ?
     ** NO

Please enter CSF findings in the following table

CSF WBC	Percent PMN	Protein	CSF Glucose	Peripheral Glucose
27)** 100	99	280	5	UNKNOWN

     28) Do you know the results of a recent CBC of Pt538?
     ** YES

Please enter CBC results in the following table

WBC in 1000's	Percent PMN	Percent BANDS
29)**_ 1.9	UNKNOWN	UNKNOWN

     30) Has Pt538 had any injury or insult to, or defect in the central-
         nervous-system (excluding neurosurgery)?
     ** NO
     31) Has Pt538 ever undergone any type of surgery?
     ** YES
     32) Does Pt538 have a history of NEUROSURGERY?
     ** NO
     33) Does Pt538 live in a crowded environment?
     ** NO
     34) Do you suspect recent alcoholic history in Pt538?
     ** NO
     35) Is meningitis a hospital-acquired infection?
     ** WHY

[i.e. WHY is it important to determine whether the infection was acquired
     while the patient was hospitalized?]

[2.0] This will aid in determining the organisms (other than those seen on
     cultures or smears) which might be causing the infection.
     It has already been established that
         [2.1] the infection which requires therapy is meningitis, and
         [2.2] organisms were not seen on the stain of this pending csf culture,
               and
         [2.3] the type of the infection is bacterial, and
         [2.4] neurosurgery is not one of the relevant items from the history
               of pt538, and
         [2.5] the age of pt538 is greater than 10 days
     Therefore, if
         [2.6] the infection was acquired while the patient was hospitalized

then
        there is evidence that the organisms (other than those seen on
            cultures or smears) which might be causing the infection is e.coli
            (.75) staphylococcus-coag-pos (.3) pseudomonas-aeruginosa (.3)
            klebsiella-pneumoniae (.5)
    Author: Yu
        [RULE545]

    [back to question 35...]
    ** HOW 2.3

[i.e. HOW was it established that the type of the infection is bacterial?]

The following rules concluded about:
    the likelihood that the type of the infection is bacterial

                    Rule   Yes   No

        [2.7]   RULE148   (.19)
        [2.8]   RULE500   (.51)
        [2.9]   RULE501   (.95)
        [3.0]   RULE502   (.97)
        [3.1]   RULE526   (.98)
        [3.2]   RULE504   (.97)
        [3.3]   RULE524   (.98)

Thus, is has been established that the type of the infection is bacterial.

    [back to question 35...]
    ** HOW 2.5

[i.e. HOW was it established that the age of pt538 is greater than 10 days?]

In answer to question 2 you said that the age of pt538 is 34 years

Thus, it has been established that the age of pt538 is greater than 10 days.

    [back to question 35...]
    ** Y
    36) Is Pt538's clinical history consistent with EPIGLOTTITIS?
    ** NO
    37) Is Pt538's clinical history consistent with OTITIS-MEDIA?
    ** NO
    38) Has Pt538 ever undergone splenectomy?
    ** NO
    39) Is Pt538 a burn patient?
    ** WHY

[i.e. WHY is it important to determine whether PATIENT-538 has been seriously
    burned?]

[4.0] This will aid in determining the organisms (other than those seen on
    cultures or smears) which might be causing the infection.

                                    120

It has already been established that
    [4.1] the infection which requires therapy is meningitis, and
    [4.2] organisms were not seen on the stain of this pending csf culture,
            and
    [4.3] the type of the infection is bacterial
Therefore, if
    [4.4] pt538 has been seriously burned
then
    there is suggestive evidence (.5) that pseudomonas-aeruginosa is one
        of the organisms (other than those seen on cultures or smears)
        which might be causing the infection
Author: Yu
    [RULE578]
  [back to question 39...]
  ** WHY

[i.e. WHY is it important to determine the organisms (other than those seen
 on cultures or smears) which might be causing the infection?]

[5.0] ... in order to determine the therapeutic regimen of PATIENT-538.
    If
        [5.1] information has been gathered about organisms isolated from the
                patient, organisms noted on smears of the patient, negative
                cultures of the patient, suspected infections without
                microbiological evidence, current drugs of the patient and
                prior drugs of the patient, and
        [5.2] an attempt has been made to deduce the organisms which require
                therapy, and
        [5.3] You have given consideration to organisms (other than those
                seen in cultures or smears) which might be present, and found
                at least one organism which requires therapy.
    then
        determine the best therapy recommendation from among the drugs likely
            to be effective against the significant organisms
    [RULE092]

  [back to question 39...]
  ** Y

I have found evidence (RULE545 RULE557) that treatment should cover for E.coli
    causing meningitis.
I have found evidence (RULE545 RULE557 RULE578) that treatment should cover
    for Pseudomonas-aeruginosa causing meningitis.
I have found evidence (RULE545 RULE557) that treatment should cover for
    Klebsiella-pneumoniae causing meningitis.
I have found evidence (RULE507) that treatment should cover for Diplococcus-
    pneumoniae causing meningitis.
    40) Does Pt538 have focal neurological signs?
    ** UNKNOWN
    [Determining which drugs are desirable for use against the E.coli...]
    [Determining which drugs are desirable for use against the Pseudomonas-
        aeruginosa...]
    [Determining which drugs are desirable for use against the Klebsiella-
        pneumoniae...]

121

[Determining which drugs are desirable for use against the Diplococcus-
     pneumoniae...]

   My therapy recommendations will be designed to treat for organisms
that either are very likely or, although less likely, would have a
significant effect on therapy selection if they were present. It is
important to cover for the following probable infection(s) and
associated organism(s):

INFECTION-1 is MENINGITIS

   + <ITEM-1> E.COLI [from clinical evidence only]
   + <ITEM-2> PSEUDOMONAS-AERUGINOSA [from clinical evidence only]
   + <ITEM-3> KLEBSIELLA-PNEUMONIAE [from clinical evidence only]
   + <ITEM-4> DIPLOCOCCUS-PNEUMONIAE [from clinical evidence only]
   41) Does Pt538 have a clinically significant allergic reaction to any
       antimicrobial agent?
   ** NO
[Considering AMPICILLIN-AND-GENTAMICIN for use against INFECTION-1...]
   42) Do you have reason to suspect that Pt538 may have impaired renal
       function?
   ** NO

   43) Pt538's weight in kilograms (or <number> POUNDS):
   ** 70

[REC-1] My preferred therapy recommendation is as follows:
   In order to cover for Items <1 2 3 4>:
        Give the following in combination:
        1) AMPICILLIN
        Dose: 3.5g (28.0 ml) q4h IV [calculated on basis of 50 mg/kg]

        2) GENTAMICIN
        Dose: 119 mg (3.0 ml, 80mg/2ml ampule) q8h IV [calculated on basis of
             1.7 mg/kg] plus consider giving 5 mg q24h Intrathecal
        Comments: Monitor serum concentrations

        Since high concentrations of penicillins can inactivate
        aminoglycosides, do not mix these two antibiotics in the same IV
        bottle.

Do you wish to see the next choice therapy?
** YES
Which, if any, of the above drugs do you wish eliminated from consideration?
** AMPICILLIN

[Considering GENTAMICIN for use against INFECTION-1...]
[Considering PENICILLIN for use against INFECTION-1...]

[REC-2] Next best therapy recommendation:

In order to cover for Items <1 2 3>:
     Give: GENTAMICIN
     Dose: 119 mg (3.0 ml, 80mg/2ml ampule) q8h IV [calculated on basis of
          1.7 mg/kg] plus consider giving 5 mg q24h Intrathecal
     Comments: Monitor serum concentrations

In order to cover for Item <4>:
     Give: PENICILLIN
     Dose: 3.5 million units (7.0 ml) q4h IV [calculated on basis of 50000
          units/kg]

RECOMMENDATION-2 is equivalent to RECOMMENDATION-1.

Would you like to see details of this comparison?
** YES
Looking at the coverage of the most likely items, we see that:

     RECOMMENDATION-1 consists of 1 first choice drug (AMPICILLIN for ITEM-1
     and ITEM-4) and 1 third choice drug (GENTAMICIN for ITEM-1, ITEM-2 and
     ITEM-3)

     RECOMMENDATION-2 consists of 1 first choice drug (PENICILLIN for ITEM-4)
     and 1 third choice drug (GENTAMICIN for ITEM-2, ITEM-1 and ITEM-3)

ORGANISMS	RECOMMENDATION-1	RECOMMENDATION-2
most likely'	Drug -- Choice Ranking	Drug -- Choice Ranking
ITEM-1	AMPICILLIN-AND-GENTAMICIN -- 1st	
		GENTAMICIN -- 3rd
ITEM-2	GENTAMICIN -- 2nd	GENTAMICIN -- 2nd
ITEM-3	GENTAMICIN -- 3rd	GENTAMICIN -- 3rd
ITEM-4	AMPICILLIN -- 1st	PENICILLIN -- 1st

(The desirability of a drug is defined to be its lowest ranking for the items
it covers.)

Do you wish to see the next choice therapy?
** NO

V. Answers to User's Questions about this Consultation and
        About the Knowledge Base

    Please ask any simple-English questions regarding the above
consultation session or the overall knowledge of the consultation
system. When discussing the consultation above, please refer to
organisms, cultures, etc. by their numeric names (eg. ORGANISM-1).
Questions which were asked during the consultation may be specified
by the numbers that preceded them above, (e.g., 'WHY DID YOU ASK
QUESTION 1?'). Control-0 will abort type-out, or 'cancel' the current
question. Type HELP for a list of the options available. Enter the
word STOP when you have no further questions.

Type HELP for the list of question options.

** HELP

    QA-OPTIONS:
    (Type ctrl-0 to abort printout)
    ADVICE - prints suggestions for using the QA system effectively
    QT - prints examples of question types recognized by the system
    EQ <numbers> - prints an explanation of the specified
        question(s) from the consultation, e.g., EQ 8 13
    LP <object type> - prints list of attributes associated
        with object type, e.g., LP CULTURE. (Object types are
        patient, infection, culture, organism, and drug.)
    PR <numbers> - prints specified rule(s), e.g., PR 373 49 87 2
    EXPLTHEP - prints a short description of the therapy program
    REC - prompts user for alternative therapy RECommendation and
            compares it to mycin's recommendation
    RX <infection> <organism identities> - prints treatment lists for
        specified organisms causing the indicated infection, e.g.,
        RX MENINGITIS ENTEROCOCCUS
    JUSTIFICATION - when rules are printed out, include their
        justifications
    LITERATURE - when rules are printed out, include their literature
        references
    NOJUST - stop giving justification and/or literature with rules
    TERSE - print rules in terse format
    VERBOSE - turns off terse printing of rules
    STOP - escape from QA-module
    HELP - prints this list

** JUSTIFICATION

** LITERATURE

** WHY DID YOU TREAT E.COLI?

ITEM-1 is E.coli associated with meningitis. I decided to treat ITEM-1 on
        the basis of clinical evidence only.

The following rules concluded about:
        the likelihood that e.coli is one of the organisms (other than those

seen on cultures or smears) which might be causing the infection

RULE545 was used to conclude that e.coli is one of the organisms (other than
     those seen on cultures or smears) which might be causing the infection
     (.74). The last question asked before the conclusion was made was 35.
RULE557 was used to conclude that e.coli is one of the organisms (other than
     those seen on cultures or smears) which might be causing the infection
     (.93). The last question asked before the conclusion was made was 37.

The parameters most frequently used in these rules were:
     the infection which requires therapy
     whether a smear of the culture was examined
     whether organisms were seen on the stain of the culture
     the type of the infection

Would you like a summary of ALL the parameters used by these rules?   ** YES

           Parameter   (rules with this parameter)

the white count from the patient's peripheral CBC (in thousands) (557)
whether the results of a recent CBC of the patient are available (557)
whether the infection was acquired while the patient was hospitalized (545)
the age of the patient (545)
the relevant items from the history of the patient (545)
the type of the infection (545, 557)
whether organisms were seen on the stain of the culture (545, 557)
whether a smear of the culture was examined (545, 557)
the infection which requires therapy (545, 557)

** WHY DIDN'T YOU GIVE TETRACYCLINE FOR E.COLI IN REC-1?
Shall I describe the therapy algorithm for you?** YES
   DESCRIPTION OF THERAPY SELECTION
           [Type cntrl-O to abort printout]

           MYCIN is designed to provide optimal therapy recommendations. The
program uses the following three steps:
           The first step ranks the drugs for each organism that requires
therapy. Rules for doing this take into account invitro (lab) sensitivities
and the patient's response to prior and current therapies. In the absence of
other information, default lab sensitivities are used. Preference is given
to current therapies if certain criteria are satisfied.
           Next, 1 or 2 drugs are selected to cover for the organisms which are
most likely to be causing the infection. As many of the "less likely"
organisms as possible are treated, but they are disregard altogether if
3 or more drugs are necessary. Throughout, preference is given to
drugs that appear higher in the treatment lists. (These lists can be
examined via the RX option.)
           Finally, the set of drugs is tested for patient factors that preclude
their administration, such as allergies, age, and renal function.
TETRACYCLINE was not prescribed for ITEM-1
TETRACYCLINE was not prescribed for ITEM-1 in RECOMMENDATION-1 because:

TETRACYCLINE was discounted for ITEM-1 because there is evidence that
this e.coli is not sensitive to it.

Would you like to see some details?** NO

** HOW DO YOU USE THE TYPE OF THE INFECTION?
The following rules use:
    the type of the infection

154, 578, 574, 572, 570, 565, 564, 559, 557, 553, 551, 549, 547, 545, 543, 541,
    539, 537, 535, 533, 531, 529, 523, 520, 519, 517, 515, 513, 511, 507, 300,
    289, 288, 287, 280, 174, 167

The parameter most frequently concluded by these rules is:
    the organisms (other than those seen on cultures or smears) which might
        be causing the infection

Would you like a summary of ALL the parameters concluded by these rules?
    ** NO

Which of the rules do you wish to see?
    ** 280

RULE280
———————

[This rule applies to all cultures, and is tried in order to find out about
    the organisms (other than those seen on cultures or smears) which might
    be causing the infection]

    If: 1) The infection which requires therapy is meningitis, and
         2) The type of the infection is fungal, and
         3) Organisms were not seen on the stain of the culture, and
         4) The patient is not a compromised host, and ❧
         5) The patient has been to an area that is endemic for coccidiomycoses,
            and
         6) The race of the patient is one of: black asian indian, and
         7) The cryptococcal antigen in the csf was not positive
    Then: There is suggestive evidence (.5) that cryptococcus is not one of the
           organisms (other than those seen on cultures or smears) which
           might be causing the infection
    Author: YU
    Justification: Dark-skinned races, especially Filipino, Asian, and Black (in
        that order) have an increased susceptibility to coccidiomycoses
        meningitis.
    Literature: Stevens, D.A. et al. Miconazole in Coccidiomycosis. Am J Med,
        60:191-202, Feb 1976.

** HOW DO YOU TREAT PNEUMCCCCCAL PNEUMONIA
The MYCIN system currently can treat the following infections: BACTEREMIA
MENINGITIS CYSTITIS. Because the knowledge base is incomplete, all other
    infections will be treated like bacteremia.

    For treatment of DIPLOCOCCUS-PNEUMONIAE in BACTEREMIA

The drugs of choice (ranked by relative efficacy, assuming sensitivity)
are:
     1st choice:
          PENICILLIN
          AMPICILLIN
     2nd choice:
          NAFCILLIN
          CEPHALOTHIN
          ERYTHROMYCIN
          CLINDAMYCIN
     3rd choice:
          TETRACYCLINE
          CHLORAMPHENICOL
          VANCOMYCIN
     Duration of therapy: 28 days

** WHAT DOSE OF ERYTHROMYCIN DO YOU RECOMMEND?
When ERYTHROMYCIN is given for BACTEREMIA, the recommended dosage is:
if age is > 12 yrs then 12 mg/kg q6h IV
     else if age is > .08 yrs then 5 mg/kg q6h IV
     else if age is > .02 yrs then 5 mg/kg q12h IV
[Different dosages are recommended for different infections.]

** HOW DO YOU TREAT CYSTITIS
In general, treatment for an infection depends on the causative organism.
Please include in your question the organism of interest.

** STOP

VI.   Dose Calculation Consultation Session

MYCIN 3-Jun-78 ...

Special options (type ? for help):
** DOSE

Instructions? (Y or N)
** YES

    This program makes dosage recommendations for a patient with an
infectious disease. You will be asked a few facts about the patient
which will be used to determine the appropriate dosages. Since the
recommended dose of an antibiotic may differ with infection, you will
first be asked to enter the infection for which the drugs are to be
given. Then you will be prompted for an antibiotic and will be told
what dosage to give. You may then request dosing information for
another antibiotic (type RETURN if you are not interested in other
drugs). You may also request dosing information for antibiotics to
treat another infection (type RETURN when you are done).

        --------PATIENT-541--------
1) Patient's name:
** OSCAR LOVEJOY
2) Age:
** 34
3) Sex:
** M
4) Do you have reason to suspect that Oscar Lovejoy may have impaired
   renal function?
** Y
5) What is the most recent creatinine clearance in ml/min which you feel
   is a true representation of Oscar Lovejoy's renal function?
** U
6) What is the most recent serum creatinine of Oscar Lovejoy (in
   mg/100ml)?
** 1.9
7) What is the previous serum creatinine of Oscar Lovejoy (in mg/100ml)?
** 1.8
8) Number of days between Oscar Lovejoy's two most recent serum
   creatinines:
** 2
9) Oscar Lovejoy's weight in kilograms (or <number> POUNDS):
** 70
10) Oscar Lovejoy's height in centimenters (or <number> INCHES):
** 175
Estimated body surface area is 1.9 sq. meters.
Dosage adjustments will be based on the calculated creatinine clearance of
         42.7 ml/min/1.73 sq. meters (adjusted to average body surface
         area.)
Infection: MENINGITIS
Drug: GENTAMICIN

                        After a loading dose of:
                             112 mg (2.8 ml, 80mg/2ml ampule) IV [calculated on basis
                                  of 1.6 mg/kg],
                        give:
                             70 mg (1.8 ml, 80mg/2ml ampule) q8h IV [calculated on
                                  basis of 1.0 mg/kg] plus consider giving 5 mg q24h
                                  Intrathecal
                   Or, after a loading dose of:
                             140 mg (3.6 ml, 80mg/2ml ampule) IV [calculated on basis
                                  of 2.0 mg/kg],
                        give:
                             119 mg (3.0 ml, 80mg/2ml ampule) q14h IV [calculated on
                                  basis of 1.7 mg/kg] plus consider giving 5 mg q24h
                                  Intrathecal
                   [normal dose is (1.7 mg/kg q8h IV plus consider giving 5 mg
                             q24h Intrathecal)]
New interval (or CR)
** 12
                        After a loading dose of:
                             133 mg (3.4 ml, 80mg/2ml ampule) IV [calculated on basis
                                  of 1.9 mg/kg],
                        give:
                             105 mg (2.6 ml, 80mg/2ml ampule) q12h IV [calculated on
                                  basis of 1.5 mg/kg] plus consider giving 5 mg q24h
                                  Intrathecal
Drug: CHLORAMPHENICOL
                   Give: 1.75g (17.6 ml) q6h IV [calculated on basis of 25 mg/kg]
Drug:
Infection:

     Please ask any simple-English questions regarding the above
consultation session or the overall knowledge of the consultation
system. When discussing the consultation above, please refer to
organisms, cultures, etc. by their numeric names (eg. ORGANISM-1).
Questions which were asked during the consultation may be specified
by the numbers that preceded them above, (e.g., 'WHY DID YOU ASK
QUESTION 1?'). Control-O will abort type-out, or 'cancel' the current
question. Type HELP for a list of the options available. Enter the
word STOP when you have no further questions.

Type HELP for the list of question options.

** JUSTIFICATION

** LITERATURE

** HOW DID YOU CALCULATE THE DOSE FOR GENTAMICIN?

When GENTAMICIN is given for MENINGITIS, the recommended dosage is:
if age is > 2 yrs then 1.7 mg/kg q8h IV plus consider giving 5 mg q24h
     Intrathecal
     else 2.3 mg/kg q8h IV plus consider giving 2.5-4 mg/day intrathecal

The normal dose for Oscar Lovejoy is: 119 mg (3.0 ml, 80mg/2ml ampule) q8h IV
    [calculated on basis of 1.7 mg/kg] plus consider giving 5 mg q24h
    Intrathecal.

GENTAMICIN is excreted by the kidneys, so its dosage must be modified in
    renal failure.

The following table shows how the patient's renal function was determined:

Identifier	Value	Definition
SCR1	1.9	the most recent serum creatinine of oscar lovejoy (in mg/100ml)
SCR2	1.8	the previous serum creatinine of oscar lovejoy (in mg/100ml)
CCR	46.9	oscar lovejoy's most recent creatinine clearance (in ml/min)
CCr(f)	42.7	oscar lovejoy's creatinine clearance, adjusted for normal body surface area (ml/min/1.73 sq. meters)
CCr(n)	100	the average normal creatinine clearance for oscar lovejoy's age group
CCr(min)	80	the minimum normal creatinine clearance for oscar lovejoy's age group

Since CCr(f) < CCr(min), PATIENT-541 has renal impairment.

To modify the dosage of GENTAMICIN, the following pharmacokinetic parameters
    were used:

Identifier	Value	Definition
a	7.0	the percentage hourly loss of a drug due to nonrenal processes
b	.22	the slope of the plot of the elimination rate constant of a drug versus the patient's creatinine clearance
Q	.57	the drug's elimination rate fraction: elimination rate of the drug as a fraction of the drug's 'normal' elimination rate.

Q was calculated using the formula:

$$Q = [a + b * CCr(f)] / [a + b * CCr(n)]$$

My first dosage recommendation decreased the amount of GENTAMICIN and left
    the dosing interval unchanged. The new amount was calculated by
    multiplying the old amount by Q.
The second recommendation increased the dosing interval and left the amount
    of GENTAMICIN unchanged. The new interval was calculated by dividing
    the old interval by Q.
** HOW DID YOU DETERMINE THE PATIENT'S BODY SURFACE AREA?

RULE104 was used to conclude that oscar lovejoy's estimated body surface area

is 1.9. The last question asked before the conclusion was made was 10.

** PR 104

RULE104
————————

[This rule applies to any patient, and is tried in order to find out about the
     patient's estimated body surface area]

   If: 1) The weight of the patient is known, and
        2) The height of the patient is known
   Then: Use Boyd's algorithm to calculate body surface area from weight and
             height [Boyd, E. The Growth of the Surface Area of the Human
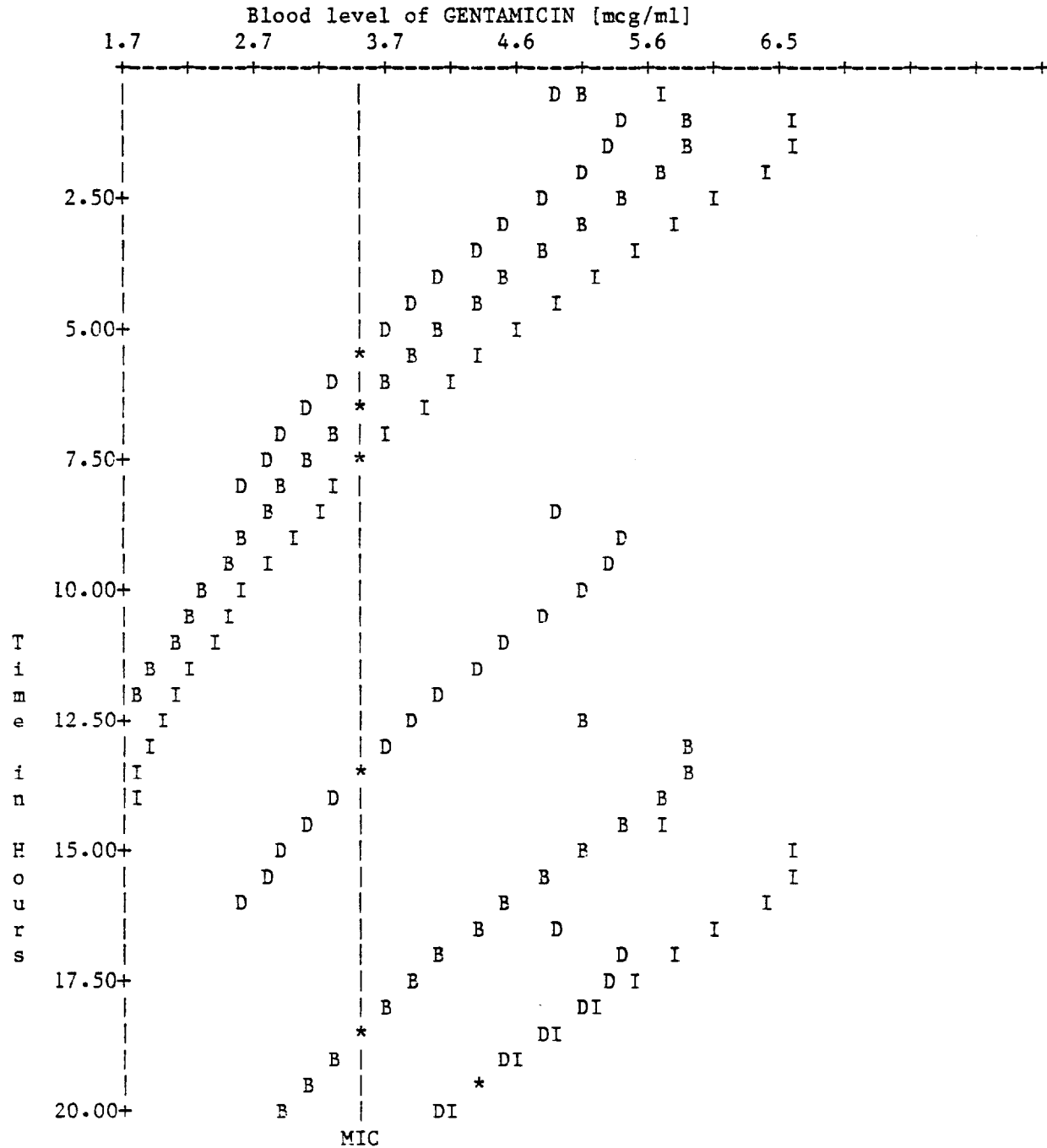   Body, 1935 pp 1 - 133.]
   Author: Wraith
   Justification: A comparative evaluation with statistical analysis of the
       Boyd method with the DuBois formula demonstrated that the Boyd formula is
       more accurate for abnormal body types. [Sendray J et.al., Determination
       of human body surface area from height and weight, Applied Physiology,
       7(1):1-12, July 1954.]
   Literature: Boyd, E, The growth of the surface area of the human body, c.
       1935, pp 1-133.

VII. Graph of Blood Levels of Anitbiotics

    The graph below provides an estimate of the steady state blood levels
of gentamicin over time for the three regimens suggested by Mycin.
A graph of this form may aid the physician in the selection of the most
appropriate regimen.

```
                 Blood level of GENTAMICIN [mcg/ml]
            1.7         2.7         3.7         4.6         5.6         6.5
           +---+---+---+---+---+---+---+---+---+---+---+---+---+---+---+
            |           |                  D B       I
            |           |                    D     B        I
            |           |                    D     B        I
            |           |                   D     B      I
      2.50+ |           |                 D     B      I
            |           |                D     B     I
            |           |              D     B     I
            |           |        D     B     I
            |           |      D     B     I
      5.00+ |         | D   B       I
            |         *     B     I
            |       D | B      I
            |     D   *     I
            |   D   B | I
      7.50+ |   D  B     *
            |  D  B   I |
            |    B   I  |                  D
            |   B   I   |                      D
            |  B   I    |                     D
     10.00+ |  B  I     |                   D
            |  B  I     |                 D
    T       | B  I      |              D
    i       | B  I      |            D
    m       |B  I       |          D
    e 12.50+ I          |        D              B
            | I         |     D                     B
    i       |I          *                         B
    n       |I        D |                      B  I
            |       D   |                    B   I
    H 15.00+|      D    |                 B              I
    o       |     D     |               B                I
    u       |    D      |             B                I
    r       |           |           B            D       I
    s       |         B |         B       D    D  I
     17.50+ |        B  |        B            D I
            |         | B           DI
            |         *         DI
            |    B  |        DI
            |   B   |      *
     20.00+ |      B   |      DI
                   MIC
```

              KEY
D   - drug level for modified dose [70.0 mgm every   8.0 hours]

I   - drug level for modified interval [119.0 mgm every 14.0 hours]
B   - drug level for modified interval and dose [98.0 mgm every 12.0 hours]
MIC - minimum inhibitory concentration [mcg/ml]
TL - toxic level [mcg/ml]
*   - indicates overlapping curves

VII.   Therapy Recommendation Consultation Session

MYCIN 3-Jun-78 ...

Special options (type ? for help):
** REC

Instructions? (Y or N)
** YES

    This program recommends therapy for a patient with infectious
diseases. You will be asked a few facts about the patient which will
be used to determine the dosages and appropriateness of giving
certain antibiotics. Then you will be prompted for the infection(s)
that you want treated and the organism(s) which you feel may be
responsible for each infecton.
    MYCIN cannot make a recommendation without knowing the
infection(s) and organism(s) to be treated. Other information is
requested for the purpose of refining therapy selection, but is not
essential; you may answer UNKNOWN to any of these questions.

        Case 2, AA, 47-24-31, VMC, A 3 year old female with one day history
of headache, lethargy, fever to 104. L.P.: Protein 25, glucose 95 110 cells
with 96% polys. Pe: Meningismus, petechiae on arms and shoulders.

28-NOV-77 11:05:51
[consultation of 23-JUN-76 12:00]

        --------PATIENT-683--------
    1) Patient's name: (first-last)
    ** PT683
    2) Age:
    ** 3.5
    3) Sex:
    ** FEMALE
    4) Is Pt683 currently receiving therapy with any antimicrobial agent?
    ** NO
    5) During the present illness, has Pt683 been treated with an
       antimicrobial agent which is no longer being administered?
    ** NO
Infection: MENINGITIS
Organism: VIRUS
        [Determining which drugs are desirable for use against the Virus...]
    Are there any other organisms which might be causing this meningitis?
** YES
Organism: HEMOPHILUS-INFLUENZAE
    6) Has the lab reported the in-vitro antibiotic sensitivities of the
       hemophilus-influenzae causing the meningitis (ITEM-2)?
    ** N
       [Determining which drugs are desirable for use against the Hemophilus-
            influenzae...]
    7) Does Pt683 have a clinically significant allergic reaction to any
       antimicrobial agent?

134

** NO
Are there any other organisms which might be causing this meningitis?
** YES
Organism: NEISSERIA-MENINGITIDIS
    8) Has the lab reported the in-vitro antibiotic sensitivities of the
       neisseria-meningitidis causing the meningitis (ITEM-3)?
    ** N
      [Determining which drugs are desirable for use against the Neisseria-
           meningitidis...]
    Are there any other organisms which might be causing this meningitis?
** NO
Are there additional infections for which you want therapy recommendations?
** NO

INFECTION-1 is MENINGITIS

    +   ...   VIRUS   Supportive therapy.
    + <ITEM-2> HEMOPHILUS-INFLUENZAE
    + <ITEM-3> NEISSERIA-MENINGITIDIS

[Considering AMPICILLIN-AND-CHLORAMPHENICOL for use against INFECTION-1...]
    9) What is the most recent creatinine clearance in ml/min which you feel
       is a true representation of Pt683's renal function?
    ** UNKNOWN
    10) What is the most recent serum creatinine of Pt683 (in mg/100ml)?
    ** UNKNOWN

    11) Pt683's weight in kilograms (or <number> POUNDS):
    ** 17.7
[REC-1] My preferred therapy recommendation is as follows:
    In order to cover for Items <2 3>:
        Give the following in combination:
        1) AMPICILLIN
        Dose: 886 mg (7.0 ml) q4h IV [calculated on basis of 50 mg/kg]
        [The dose should be modified in SEVERE renal failure. I can adjust
            the dose only if a creatinine clearance or a serum creatinine
            is obtained, and the patient's weight is known.]

        2) CHLORAMPHENICOL
        Dose: 443 mg (4.4 ml) q6h IV [calculated on basis of 25 mg/kg]

    Please ask any simple-English questions regarding the above
consultation session or the overall knowledge of the consultation
system. When discussing the consultation above, please refer to
organisms, cultures, etc. by their numeric names (eg. ORGANISM-1).
Questions which were asked during the consultation may be specified
by the numbers that preceded them above, (e.g., 'WHY DID YOU ASK
QUESTION 1?'). Control-O will abort type-out, or 'cancel' the current
question. Type HELP for a list of the options available. Enter the
word STOP when you have no further questions.

Type HELP for the list of question options.

** JUSTIFICATION

** LITERATURE

** WHY DID YOU GIVE AMPICILLIN AND CHLORAMPHENICOL IN REC-1?

Shall I describe the therapy algorithm for you?** NO

AMPICILLIN-AND-CHLORAMPHENICOL was not prescribed for ITEM-1 or ITEM-3
For ITEM-1 no drug was prescribed.
For ITEM-3 AMPICILLIN was prescribed.
AMPICILLIN-AND-CHLORAMPHENICOL was prescribed for ITEM-2 in RECOMMENDATION-1
because:

> — AMPICILLIN-AND-CHLORAMPHENICOL is used for treating hemophilus-
> influenzae in meningitis
> — AMPICILLIN-AND-CHLORAMPHENICOL was considered for treating ITEM-2
> because it is more appropriate than AMPICILLIN for this patient. [RULE066]
> — ITEM-2 is sensitive to AMPICILLIN-AND-CHLORAMPHENICOL
> — there were no contraindications for it
> — it was part of the recommendation which covers for all of the items,
> using the fewest number of drugs.

Would you like to see some details?** YES

The following decision(s) are relevant:
> The drugs considered for the hemophilus-influenzae ranked according to
> desirability are ampicillin-and-chloramphenicol (.95) chloramphenicol
> (.89) carbenicillin (.79) tetracycline (.76) erythromycin (.75)
> cephalothin (.7) [RULE101 RULE066]
>
> The drugs to which the hemophilus-influenzae is sensitive are ampicillin-
> and-chloramphenicol (1.0) chloramphenicol (1.0) carbenicillin (1.0)
> tetracycline (1.0) erythromycin (1.0) cephalothin (1.0) [RULE098 RULE445]
> For the conclusion concerning ampicillin-and-chloramphenicol see RULE445.
>
> The meningitis is one of the infections for which ampicillin-and-
> chloramphenicol is appropriate [RULE441]

** PR 66

RULE066

> If: 1) Ampicillin is one of the drugs considered for the item ranked
>   according to desirability, and
>   2) The diagnosis of the infection is meningitis, and
>   3) The age of the patient is greater than or equal to 1 week, and
>   4) The identity corresponding to the item is hemophilus-influenzae, and
>   5) The item is one of the most likely (top cluster) identities causing
>   an infection in the patient, and
>   6) It is not known whether ampicillin is one of the drugs to which the
>   item is sensitive in vitro, and
>   7) A: The patient is not allergic to one or more antibiotics, or
>   B: Chloramphenicol is not one of the drugs to which the patient is
>   allergic

Then: It is definite (1.0) that the combination of ampicillin with
       chloramphenicol is a more appropriate therapy for use against the
       item
Author: Wraith
Justification: Due to the increasing number of Ampicillin-resistant
    Hemophilus influenzae isolated in the last few years a combination of
    ampicillin and chloramphenicol is recommended until sensitivities of the
    organism are known. At that time, chloramphenicol should be discontinued
    unless the organism is ampicillin-resistant, in which case, ampicillin is
    discontinued.
Literature: 1. Katz SL: Ampicillin-resistant Hemophilus influenzae type B:
    A status report. Pediat. 55:66, 1975.
        2. Katz SL et.al.: Ampicillin-resistant strains of Hemophilus
    influenzae type b, Pediat. 55:145, 1975.
        3. Nelson JD: Should ampicillin be abandoned for treatment of
    Hemophilus influenzae disease? JAMA 229:322, 1974.

** REC
For each item in turn, enter the drug you would have prescribed in
RECOMMENDATION-1.
In order to minimize the number of drugs in your recommendation, you may not
want to prescribe therapy for every item. Items which represent the most
likely organisms are indicated with a plus sign (+).

 + ITEM-1 -- the virus **
 + ITEM-2 -- the hemophilus-influenzae ** CARBENICILLIN
 + ITEM-3 -- the neisseria-meningitidis ** CARBENICILLIN

[Checking for contraindications...]
[Considering CARBENICILLIN for use against INFECTION-1...]
[No contraindications found...]

[Now comparing your prescription to MYCIN's...]

ORGANISMS   MYCIN's regimen   Your regimen
"most likely" Drug -- Choice Ranking   Drug -- Choice Ranking
   ITEM-2   AMPICILLIN-AND-CHLORAMPHENICOL -- 1st
                                             CARBENICILLIN -- 3rd
   ITEM-3   AMPICILLIN -- 1st   CARBENICILLIN -- 2nd

(The desirability of a drug is defined to be its lowest ranking for the items it covers.)

Your recommendation for the most likely organisms(s) is not optimal, since your prescription of 1 third choice drug (CARBENICILLIN for ITEM-3 and ITEM-2) should be avoided.

[You may refer to your regimen as RECOMMENDATION-2 in later questions.]

** STOP

IX.   Running a Consultation of a Stored Patient in Summary Form
           with Rule Acquisition to Correct Diagnosis

         Case 10, TS, VMC, 9-49-13-77, A 73 year old female with rheumatoid
arthritis, congestive heart failure, and gastritis on chronic prednisone 15
mg/day and coumadin. Admitted for a GI bleed. Progressive obtundation began
while in hospital. LP revealed: protein 158, glucose 20, 38 cells with 34 %
polys, 66% lymphs. Gram stain and India Ink prep revealed budding yeast-like
cells. Treatment: Begun on Amphotericin B IV and IT as well as 5-fc. Final
dx: Cryptococcal meningitis.

29-NOV-77 01:45:12
[consultation of 9-OCT-76 12:00]

    Pt709 is a 73 year old female, caucasian.
    Patient-709 is not an alcoholic.
    Patient-709 is a compromised host.
    Patient-709 is immunosuppressed.
    Patient-709 does not live in a crowded environment.

    Past Medical History:
    Patient-709 is not allergic to one or more antibiotics.
    Patient-709 has not undergone surgery.
    Patient-709 does not have a tb risk factor.
    Patient-709 has not recently been exposed to a contagious disease.

    Recent Medical History:
    The csf has not been tested for cryptococcus antigen.
    Patient-709 has not shown symptoms of mumps.
    Otitis-media is not one of the diagnoses which are consistent with the
    patient's clinical history.
    Epiglottitis is not one of the diagnoses which are consistent with the
    patient's clinical history.
    Patient-709 has not had an injury or insult to, or defect in the CNS.
    Patient-709 has had recent neurologic signs.
    The duration of the neurological signs is 4 days.
    Patient-709 has had recent neurologic symptoms.
    The duration of the neurological symptoms is 2 days.

    Physical:
    The weight of PATIENT-709 is 68.1 kgms.
    The height of PATIENT-709 is 165.1 cms.
    Patient-709 is febrile.
    Patient-709 has not been seriously burned.
    Patient-709 does not have a rash or cutaneous lesions.
    Patient-709 has not focal neurological signs.
    Patient-709 does not have evidence of ocular nerve dysfunction.
    Patient-709's X-ray is abnormal.
    Active-tb is not one of the diseases which the patient's chest X-ray
    suggests.
    Lobar-pneumonia is not one of the diseases which the patient's chest X-ray
    suggests.

CULTURE-1: This pending csf culture was drawn on 9-OCT-76 12:00.
A smear of this pending csf culture was examined.

--------ORGANISM-1--------
The stain of ORGANISM-1 is indiaink.
The CSF cultures will be considered to be associated with MENINGITIS.
The number of days since clinical evidence of the infection was first
noted is 4 days.
[Considering organisms (other than those isolated from cultures) which might
be present...]
Lymphoma is not one of the diagnoses of the patient.
Leukemia is not one of the diagnoses of the patient.
The CSF protein is 158.
The WBC from the CSF is 38.
The percent of pmns in the csf is 34.
The csf glucose value is 20.
The simultaneous blood glucose value is 95.
The results of a recent CBC of pt709 are available.
The white count from pt709's peripheral CBC (in thousands) is 9.4.
Pt709 has received corticosteroids just prior to or during this infection.
The infection was acquired while the patient was hospitalized.

I have found evidence (RULE564) that treatment should also cover for VIRUS
causing MENINGITIS.
[Determining which drugs are desirable for use against the virus...]

My therapy recommendations will be designed to treat for organisms
that are either very likely or, although less likely, would have a
significant effect on therapy selection if they were present. It is
important to cover for the following probable infection(s) and
associated organism(s):

INFECTION-1 is MENINGITIS

   +   ...   VIRUS [from clinical evidence only]   Supportive therapy.

   Please repeat the LP in 8 - 24 hours for cultures (bacterial, fungal,
   Tb, and viral), cells, and chemistries.

   Blood cultures should be drawn.

Is this correct?
** NO
I was unable to make any conclusion about the identity of ORGANISM-1

Is that correct?
** NO
Should it have been possible to conclude a value?
** YES
Which one(s)?
** CRYPTOCOCCUS
Let's look at the CRYPTOCOCCUS ...

Do you want to
       1] track down the problem yourself, or
       2] allow me to guide the debugging, or
       3] ignore it for now and fix it another time?
Type 1, 2 or 3
** 2
The following applicable rule(s) were unable to conclude about whether the
identity of ORGANISM-1 is cryptococcus

RULE378

  If: 1) The infection which requires therapy is meningitis, and
       2) The stain of the organism is acidfast
  Then: 1) There is suggestive evidence (.7) that the identity of the
            organism is mycobacterium-tb, and
         2) There is evidence that the organisms (other than those seen on
            cultures or smears) which might be causing the infection is
            coccidioides (-.8) cryptococcus (-.8)

Clause 2 of RULE378 [the stain of the organism is acidfast] was already known
      to be false for ORGANISM-1, so the rule was never tried.

Is this rule correct?
** YES
Should its premise have succeeded for this case?
** NO

Is there a rule missing?
** YES
Ok, then I need a rule that would allow me to deduce that the identity of
ORGANISM-1 is cryptococcus
Would you like to give me such a rule now?
** YES
Good...
The new rule will be called RULE260

If: 1 - THE INFECTION WHICH REQUIRES THERAPY IS MENINGITIS
     and 2 - THE STAIN IS INDIAINK
     and 3 -
Then: 1 - THE IDENTITY IS CRYPTOCOCCUS (.7)
     and 2 -
This may take a few moments.

This is my understanding of your rule:
RULE260

  If: 1) The infection which requires therapy is meningitis, and
       2) The stain of the organism is indiaink
  Then: There is suggestive evidence (.7) that the identity of the organism
            is cryptococcus

Okay? (Yes or No)
** YES

141

This rule may be best described as a rule which:
1 - applies to all organisms
2 - applies to organisms from positive cultures
3 - applies to organisms seen on smears of specimens for pending cultures
** 1

Please type your last name- BENNETT

Please describe briefly why you added (or changed) this rule.
Type control-Z when done.
** THE SYSTEM FAILED TO CONCLUDE CRYPTOCOCCUS FOR AN INDIAINK STAIN AND
** MENINGITIS AS THE INFECTION
**
RULE260 has now been added to the knowledge base.

I will now re-run the consultation to check the effectiveness of your new
     rule. Get comfortable, this may take a few moments.

29-NOV-77 01:53:52
[consultation of 9-OCT-76 12:00]

     --------PATIENT-709--------

        --------CULTURE-1--------

           --------ORGANISM-1--------

The CSF cultures will be considered to be associated with MENINGITIS.
[Considering organisms (other than those isolated from cultures) which might
be present...]

I have found evidence (RULE564) that treatment should also cover for VIRUS
     causing MENINGITIS.
     [Determining which drugs are desirable for use against the cryptococcus...]
     [Determining which drugs are desirable for use against the virus...]

     My therapy recommendations will be designed to treat for organisms
that are either very likely or, although less likely, would have a
significant effect on therapy selection if they were present. It is
important to cover for the following probable infection(s) and
associated organism(s):

INFECTION-1 is MENINGITIS

  + <ITEM-1> CRYPTOCOCCUS [ORGANISM-1]
        ...   VIRUS [from clinical evidence only]   Supportive therapy.

[My recommendations will cover for all of the most likely items (indicated
by a "+" above). Less likely items will be covered unless this would

require an excessive number of drugs.]

Well, it appears that things are straightened out.

Is this correct now?
** YES
Good, then the problem's been fixed.
      ...continuing with therapy selection now.

[Considering 5-FLUOROCYTOSINE-AND-AMPHOTERICIN-B for use against INFECTION-1...]
Estimated body surface area is 1.8 sq. meters.
Dosage adjustments will be based on the calculated creatinine clearance of
          85.4 ml/min/1.73 sq. meters (adjusted to average body surface
          area.)
Normal doses of all drugs will be given

[REC-1] My preferred therapy recommendation is as follows:
          Give the following in combination:
          1) 5-FLUOROCYTOSINE
          Dose: 2.56g (10 250mg-capsules) q6h PO [calculated on basis of 37.5
              mg/kg]
          Comments: Gradually increase dose from 12 mg/kg q6h to the above dose
              as tolerated.

          2) AMPHOTERICIN-B
          Dose: 20 mg (200.0 ml) q24h IV [calculated on basis of .3 mg/kg]
          Comments: Begin with 1 mg IV on day 1, 5 mg on day 2, 10 mg on day 3,
              and then gradually increase dose by 5-10 mg/day until the above
              dose is administered. Infuse slowly over 3-6 hrs. Plus,
              consider giving 0.5mg intrathecally every other day. Greater
              than 15 mg intrathecally is rarely indicated.

     Please obtain samples for fungal, TB, and viral cultures, cytology, VDRL
     (blood and CSF), coccidiodes complement-fixation (blood and CSF),
     cryptococcal antigen (blood and CSF), viral titers (blood and CSF). An
     infectious disease consult may be advisable.

     Blood cultures should be drawn.

APPENDIX B:
KNOWLEDGE ENGINEERING FOR MEDICAL DECISION MAKING:
A Review of Computer-Based Clinical Decision Aids

1   INTRODUCTION

As early as the 1950's it was recognized that computers could conceivably assist with clinical decision making [57], and both physicians and computer scientists began to analyze medical diagnosis with a view to the potential role of automated decision aids in that domain [55]. A variety of techniques have been applied to computer-aided clinical decision making, accounting for at least 800 references in the clinical and computing literature [104]. In this article we review several methodologies and attempt to identify the important issues that account for both the multiplicity of approaches to the problem and the limited clinical success of most of the systems developed to date. Although there have been previous reviews of computer-aided diagnosis [42], [86], [106], our emphasis here will be somewhat different. We will focus on the representation and utilization of knowledge, termed "knowledge engineering," and the inadequacies of data-intensive techniques which have led to the exploration of novel symbolic reasoning approaches during the last decade.

1.1   Reasons For Attempting Computer-Aided Medical Decision Making

It is generally recognized that accelerated growth in medical knowledge has necessitated greater sub-specialization among physicians and more dependence upon assistance from other experts when a patient presents with a complex problem outside one's own area of expertise. The primary care physician who sees the patient initially has thousands of tests available with a wide range of costs (both fiscal and physical) and potential benefits (i.e., arrival at a correct diagnosis or optimal therapeutic management). Even the experts in a field may reach very different decisions regarding the management of a specific case [122]. Diagnoses that are made, and upon which therapeutic decisions are based, have been shown to vary widely in their accuracy [22],[77], [83]. Furthermore, medical decision making has traditionally been learned by medical students in an unstructured way, largely through observing and emulating the thought processes they perceive to be used by their clinical mentors [48].

Thus the motivations for attempts to understand and automate the process of

clinical decision making have been numerous [106]. They are directed both at diagnostic models and at assisting with patient management decisions. Among the reasons for attempting such work are the following:

(1) To improve the accuracy of clinical diagnosis through approaches that are systematic, complete, and able to utilize data from diverse sources;

(2) To improve the reliability of clinical decisions by avoiding unwarranted influences of similar but not identical cases (a common source of bias among physicians), and by making the criteria for decisions explicit, and hence reproducible;

(3) To make the selection of tests and therapies efficient in that optimal decisions are reached while the expense of time or funds is minimized before definitive action is taken;

(4) To improve our understanding of clinical decision making, both so that future physicians can have better teaching in this area, and so that the computer programs we develop will be more effective and easier to understand by the physicians for whom they are designed.

1.2   The Distinction Between Data And Knowledge

The models on which computer systems base their clinical advice range from data-intensive to knowledge-intensive approaches. If there is a chronology to the field over the last 20 years, it is that there has been progressively less dependence on "pure," observational data and more emphasis on higher-level symbolic knowledge inferred from primary data. We include with domain knowledge a category of "judgmental knowledge" which reflects the experience and opinions of an expert regarding an issue about which the formal data may be fragmentary or nonexistent. Since many decisions made in clinical medicine depend upon this kind of judgmental expertise, it is not surprising that investigators should begin to look for ways to capture and utilize the knowledge of experts in decision making programs. Another reason to move away from purely data-intensive programs is that in medicine the primary data available to decision makers are far from objective [16]. They include subjective reports from patients, and error-prone observations [23]. Also, the terminology used in the reports is not standardized [7] and the classifications often overlap. Thus decision making aids must be knowledgeable about the unreliability of the data as well as the uncertainty of the inference.

For example, data-intensive programs include medical record systems which accumulate large databanks to assist with decision making. There is little knowledge per se in the databank, but there <u>are</u> large amounts of data which can help with decisions and be analyzed to provide new knowledge. A program that retrieves a patient's record for review, or even one that retrieves the records of several patients matching some set of descriptors, is performing a data management task with minimal "knowledge engineering" involved [32], [80]. On the other hand, there is knowledge contained in the conditional probabilities generated from such a databank and utilized for Bayesian analysis. At the other extreme are systems that attempt to understand and utilize the kind of expert knowledge which cannot be easily gleaned from databanks or literature reviews [69], [95]. Systems that model human reasoning or emphasize education of users tend to fall towards this end of the data-knowledge continuum.

We use the term "knowledge engineering", then, to refer to computer-based symbolic reasoning issues such as knowledge representation, acquisition, and explanation [15]. It is along these dimensions that the programs differ most sharply from conventional calculations. For example, these programs can solve problems by pursuing a line of reasoning; the individual inference steps and the whole chain of reasoning may also form the basis for explanations of decisions. A major concern in knowledge engineering is clear separation of the medical knowledge in a program from the inference mechanism that applies that knowledge to individual cases. One goal of this paper is to identify, in the strengths and weaknesses of earlier work, those issues which have motivated several current research groups to investigate the knowledge engineering approach to the automation of clinical decision aids.

1.3 Parameters For Assessing Work In The Field

The barriers to successful implementation of computer-based diagnostic systems have been analyzed on several occasions [7],[19],[99] and these need not be reviewed in detail here. However, in assessing programs it is pertinent to examine several parameters that affect the success and scope of a particular system in light of its intended users and application:

(1) How accurate is the program?[1]

---

[1]Although this is important it is not the only measure of clinical effectiveness. For example, the effects on morbidity, mortality, and length of hospital stay may also be important parameter. As we shall show, few systems have reached a stage of implementation where these parameters could be assessed.

(2) What is the nature of the knowledge in the system and how is it generated or acquired?

(3) How is the clinical knowledge represented, and how does it facilitate the performance goals of the system described?

(4) How are knowledge and clinical data utilized and how does this impact on system performance?

(5) Is the system accepted by the users for whom it is intended? Is the interface with the user adequate? Does the system function outside of a research setting and is it suitable for dissemination?

(6) What is the size of the required computing resource?

(7) What are the limitations of the approach?

One issue we have chosen not to address is the cost of a system. Not only is information on this question scanty for most of the programs, but expenses generated in a research and development environment do not realistically reflect the costs one would expect from a system once it is operating for service use.

1.4 Overview Of This Paper

An exhaustive review of computer-aided diagnosis will not be attempted in light of the vastness of the field, and we have therefore chosen to review the methodologies by discussing several representative examples of systems that have been described. The seven principal examples we have selected are not necessarily the best nor the most successful; however, they illustrate the issues we wish to discuss and encompass most of the major methodologies that have been applied to computer-based medical decision making. In several cases we have referenced other closely related systems, and the bibliography should therefore guide the reader who wishes to pursue a particular topic more thoroughly. Any attempt to categorize programs in this way is inherently fraught with problems in that several systems appropriately lay claim to more than one methodology. Thus we have occasionally felt obligated to simplify a topic for clarity in light of the overall purposes of this review and the limitations of the space available to us.

Finally, certain kinds of decision making tools have been intentionally deleted from discussion here. These include medical systems that are designed primarily for use by researchers [35], [45], [59], [84]; advanced automated instrumentation techniques such as computerized tomography[2]; signal processing

---

[2]See Kak's article in this issue of the PROCEEDINGS.

techniques such as programs for EKG analysis [73] or patient monitoring [108]; and programs designed largely for data storage and retrieval with the actual analysis and decision making left largely to the clinician [32],[52],[116]. We have also chosen to discuss working computer programs rather than theories suitable for automation or early reports of work in progress.

2   Clinical Algorithms and Automation

2.1   Overview

Clinical algorithms, or protocols, are structured decision making flowcharts to which a diagnostician or therapist can refer when deciding how to manage a patient with a specific clinical problem [90]. In general these algorithms have been designed by expert physicians for use by physicians' assistants or nurse practitioners who are substituting for physicians in the performance of certain routine clinical-care tasks[3]. The methodology has been developed in part because of a desire to define basic medical logic concisely so that detailed training in pathophysiology would not be necessary for ancillary practitioners. Experience has shown that intelligent high school graduates, selected in large part because of poise and warmth of personality, can provide excellent care guided by protocols after only 4-8 weeks of training. This care has been shown to be equivalent to that given by physicians for the same limited problems, and to be accepted by physicians and patients alike for such diverse clinical situations as diabetes management [51], [60], pharyngitis [34], headache [33], and other disease categories [97], [103].

The role of the computer in such applications has been limited, however. In fact, several groups initially experimented with computer representation of the algorithms but have since abandoned the efforts and resorted to prepared paper forms [51], [103]. In these cases the computer had originally guided the physician assistant's collection of data and had specified precisely what decisions should be made or actions taken, in accordance with the clinical algorithm. However, since the algorithmic logic is generally simple, and can often be represented on a single sheet of paper, the advantages of an automated approach over a manual system have not been clearly demonstrated. In one study

_____

[3]Clinical algorithms have also been prepared for use by physicians themselves, but Grimm has found that they are generally less well-accepted by doctors [34]. He showed, however, that physician performance could improve when protocols were used in certain settings.

Vickery showed that, although the computer system entirely eliminated errors in data collection (since the program demanded all relevant data at the appropriate time), supervising physicians could detect no significant difference between the performance of physicians' assistants using automated versus manual systems [103]. Furthermore, the computer could not, of course, decide whether the actual observations entered by the physicians' assistant were correct; yet this kind of inaccuracy was one of the most common reasons that supervisors occasionally found an assistant's performance unsatisfactory.

There are two other ways in which the computer has been utilized in the setting of clinical algorithms. One has been in the use of mathematical techniques to analyze signs and symptoms of diseases and thereby to identify those that should most appropriately be referenced in a clinical algorithm that is being prepared for the management of that disease [26],[50],[105]. The process for distilling expert knowledge in the form of a clinical algorithm can be an arduous and imperfect one [90]; formal techniques to assist with this task may prove to be very valuable.

Finally, some researchers in this area continue to use computers to assist with audit of performance by comparing actual actions taken by a physicians' assistant with those recommended by the algorithm itself. Sox et al. [97] have described a system in which the assistant's checklist for a patient encounter was sent to a central computer and analyzed for evidence of deviation from the accepted protocol. Computer-generated reports then served as feedback to the physicians' assistant and to the supervising physician.

2.2 Example

We have selected for discussion a project that differs from those previously cited in that (1) computer techniques are still being utilized, and (2) the clinical algorithms are designed for use by primary care physicians themselves. This is the cancer chemotherapy system developed in Alabama by Mesel et al. [64]. The algorithms were developed in response to a desire to allow private practitioners, at a distance from the regional tertiary-care center, to manage the complex chemotherapy for their cancer patients, without routinely referring them to the central oncologists. Mesel et al. have described a "consultant-extender system" that enables the primary physician to treat patients with Hodgkin's Disease under the supervision of a regional specialist. Five oncologists developed a care protocol for the treatment of

Hodgkin's Disease, and this algorithm was placed on-line. Once patients had been entered in the study, their private physicians would prepare encounter forms at the time of each office visit. These forms would document pertinent interval history, physical findings, and lab data, as well as chemotherapy administered. The form would then be sent to the regional center where it was analyzed by the computer and a customized clinical algorithm was produced to assist the private physician with the management of _that_ patient during the next appointment. Thus the computer program would take into account the ways in which the individual patient's disease might progress or improve and would prepare an appropriate clinical algorithm. This protocol was sent back to the physician in time for it to be available at the next office visit. The private practitioner was encouraged to call the regional specialist directly if the protocol seemed in some way inadequate or additional questions arose. The authors present data suggesting that their system was well-accepted by physicians and patients, and that excellent care was delivered. This is an interesting result in light of Grimm's experience [34]. Perhaps physicians were more accepting of the algorithmic approach in Mesel's case because it allowed them to perform tasks that they would previously not have been able to undertake _at all_. Retrospective review of cases that were treated _at the referral center_, but without the use of the protocols, showed a 16% rate of variance from the management guidelines specified in the algorithms; there was no such variance when the protocols were utilized directly. Thus algorithms may be effective tools for the administration of complex specialized therapy in circumstances such as those described.

2.3   Discussion of the Methodology

Although clinical algorithms are among the most widespread and accepted of the decision aids described in this article, the simplicity of their logic makes it clear why the technique cannot be effectively applied in most medical domains. Decision points in the algorithms are generally binary (i.e., a given sign or symptom is or is not present), and there tend to be many circumstances that can arise for which the user is advised to consult the supervising physician (or specialist). Thus the complex decision tasks are left to experts, and there is generally no formal algorithm for managing the case from that point on. It is precisely the simplicity of the algorithmic logic, and the supervising expert "escape valve", which has permitted many algorithms to be

represented on one or two sheets of paper and has obviated the need for direct
computer use in most of the systems. The contributions of clinical algorithms
to the distribution and delivery of health care, to the training of paramedics,
and to quality care audit, have been impressive and substantial. However, the
methodology is not suitable for extension to the complex decision tasks to be
discussed in the following sections.

3 Databank Analysis for Prognosis and Therapy Selection

3.1 Overview

Automation of medical record keeping and the development of computer-based
patient databanks have been major research concerns since the earliest days of
medical computing. Most such systems have attempted to avoid direct interaction
between the computer and the physician recording the data, with the systems of
Weed [115], [116] and Greenes [32] being notable exceptions. Although the
earliest systems were designed merely as record-keeping devices, there have been
several recent attempts to create programs that could also provide analyses of
the information stored in the computer databank. Some early systems [32], [47]
had retrieval modules that identified all patient records matching a Boolean
combination of descriptors; however, further analyses of these records for
decision making purposes was left to the investigator. Weed has not stressed an
analytical component in his automated problem-oriented record [116], but others
have developed decision aids which use medical record systems fashioned after
his [96].

The systems for databank analysis all depend on the development of a
complete and accurate medical record system. If such a system is developed, a
number of additional capabilities can be provided: (1) correlations among
variables can be calculated, (2) prognostic indicators can be measured, and (3)
the response to various therapies can be compared. A physician faced with a
complex management decision can look to such a system for assistance in
identifying patients in the past who had similar clinical problems and can then
see how those patients responded to various therapies. A clinical investigator
keeping the records of his study patients on such a system can utilize the
program's statistical capabilities for data analysis. Hence, although these
applications are inherently data-intensive, the kinds of "knowledge" generated
by specialized retrieval and statistical routines can provide valuable

assistance for clinical decision makers. For example, they can help physicians avoid the inherent biases that result when the individual practitioner bases his decisions primarily on his own anecdotal experience with one or two patients having a rare disease or complex of symptoms.

There are many excellent programs in this category, one of which is discussed in some detail in the next section. Several others warrant mention, however. The HELP System at the University of Utah [109], [111], [112] utilizes a large data file on patients in the Latter-Day Saints Hospital. Clinical experts formulate specialized "HELP sectors" which are collections of logical rules that define the criteria for a particular medical decision. These sectors are developed by an interactive process whereby the expert proposes important criteria for a given decision and is provided with actual data regarding that criterion based on relevant patients and controls from the computer databank. The criteria in the sector are thus adjusted by the expert until adequate discrimination is made to justify using the sector's logic as a decision tool[4]. The sectors are then utilized for a variety of tasks throughout the hospital.

Another system of interest is that of Feinstein et al. at Yale [17]. They had specific patient management decisions in mind when they developed their interactive system for estimating prognosis and guiding management in patients with lung cancer. Similarly, Rosati et al. have developed a system at Duke University which utilizes a large databank on patients who have undergone coronary arteriography [82]. New patients can be matched against those in the databank to help determine patient prognosis under a variety of management alternatives.

3.2 Example

One of the most successful projects in this category is the ARAMIS system of Fries [20]. The approach was designed originally for use in an outpatient rheumatology clinic, but then broadened to a general clinical database system (TOD) [118], [119] so that it became transferable to clinics in oncology, metabolic disease, cardiology, endocrinology, and certain pediatric subspecialties. All clinic records are kept in a flow-charting format in which a column in a large table indicates a specific clinic visit and the rows indicate the relevant clinical parameters that are being followed over time.

---

[4]This process might be seen as a tool to assist with the formulation of clinical algorithms as discussed in the previous section. Another approach using databank analysis for algorithm development is described in [26].

These charts are maintained by the physicians seeing the patient in clinic, and the new column of data is later transferred to the computer databank by a transcriptionist; in this way time-oriented data on all patients are kept current. The defined database (clinical parameters to be followed) is determined by clinical experts, and in the case of rheumatic diseases has now been standardized on a national scale [36].

The information in the databank can be utilized to create a prose summary of the patient's current status, and there are graphical capabilities which can plot specific parameters for a patient over time [118]. However, it is in the analysis of stored clinical experience that the system has its greatest potential utility [21]. In addition to performing search and statistical functions such as those developed in databank systems for clinical investigation [45], [59], ARAMIS offers a prognostic analysis for a new patient when a management decision is to be made. Using the consultative services of the Stanford Immunology Division, an individual practitioner may select clinical indices for his patient that he would like matched against other patients in the databank. Based on 2 to 5 such descriptors, the computer locates relevant prior patients and prepares a report outlining their prognosis with respect to a variety of endpoints (e.g., death, development of renal failure, arthritic status, pleurisy, etc.). Therapy recommendations are also generated on the basis of a response index that is calculated for the matched patients. A prose case analysis for the physician's patient can also be generated; this readable document summarizes the relevant data from the databank and explains the basis for the therapeutic recommendation.

The rheumatologic databank generated under ARAMIS has now been expanded to involve a national network of immunologists who are accumulating time-oriented data on their patients. This national project seeks in part to accumulate a large enough databank so that groups of retrieved patients will be sizable and thus control for some observer variability and make the system's recommendations more statistically defensible.

3.3   Discussion of the Methodology

The databank analysis systems described have powerful capabilities to offer to the individual clinical decision maker. Furthermore, medical computing researchers recognize the potential value of large databanks in supporting many of the other decision making approaches discussed in subsequent sections. There

are important additional issues regarding databank systems, however, which are discussed below.

(1) Data acquisition remains a major problem. Many systems have avoided direct physician-computer interaction but have then been faced with the expense and errors of transcription. The developers of one well accepted record system still express their desire to implement a direct interface with the physician for these reasons, although they recognize the difficulties encountered in encouraging hands-on use of a computer system by doctors [100].

(2) Analysis of data in the system can be complicated by missing values that frequently occur, outlying values, and poor reproducibility of data across time and among physicians.

(3) The decision aids provided tend to emphasize patient management rather than diagnosis. Feinstein's system [17] is only useful for patients with lung cancer, for example, and the ARAMIS (TOD) prognostic routines, which are designed for patient management, assume that the patient's rheumatologic diagnosis is already known.

(4) There is no formal correlation between the way expert physicians approach patient management decisions and the way the programs arrive at recommendations. Feinstein and Koss felt that the acceptability of their system would be limited by a purely statistical approach, and they therefore chose to mimic human reasoning processes to a large extent [53], but their approach appears to be an exception.

(5) Data storage space requirements can be large since the decision aids of course require a comprehensive medical record system as a basic component.

Slamecka has distinguished between structured and empirical approaches to clinical consulting systems [96], pointing out that databanks provide a largely empirical basis for advice whereas structured approaches rely on judgmental knowledge elicited from the literature or the minds of experts. It is important to note, however, that judgmental knowledge is itself based on empirical information. Even the expert "intuitions" that many researchers have tried to capture are based on that expert practitioner's own observations and "data collection" over years of experience. Thus one might argue that large, complete, and flexible databanks could form the basis for large amounts of judgmental knowledge that we now have to elicit from other sources. Some researchers have indicated a desire to experiment with methods for the automatic generation of medical decision rules from databanks, and one component of the

research on Slamecka's MARIS system is apparently pointed in that direction [96]. Indeed, some of the most exciting and practical uses of large databanks may be found precisely at the interface with those knowledge engineering tasks that have most confounded researchers in medical symbolic reasoning [5].

4   Mathematical Models of Physical Processes

4.1   Overview

Pathophysiologic processes can be well-described by mathematical formulae in a limited number of clinical problem areas. Such domains have lent themselves well to the development of computer-based decision aids since the issues are generally well-defined. The actual techniques used by such programs tend to reflect the details of the individual applications, the most celebrated of which have been in pharmacokinetics (specifically digitalis dosing), acid-base/electrolyte disorders, and respiratory care [63].

One or two cooperating experts in the field generally assist with the definition of pertinent variables and the mathematical characterization of the relationships among them. Often an interactive program is then developed which requests the relevant data, makes the appropriate computations, and provides a clinical analysis or recommendation for therapy based upon the computational results. Some of the programs have also involved branched-chain logic to guide decisions about what further data are needed for adequate analysis[5].

Programs to assist with digitalis dosing have progressed to the inclusion of broader medical knowledge over the last ten years. The earliest work was Jelliffe's [43] and was based upon his considerable experience studying the pharmacokinetics of the cardiac glycosides. His computer program used mathematical formulations based on parameters such as therapeutic goals (e.g., desired predicted blood levels), body weight, renal function, and route of administration. In one study he showed that computer recommendations reduced the frequency of adverse digitalis reactions from 35% to 12% [44]. Later, another group revised the Jelliffe model to permit a feedback loop in which the digitalis blood levels obtained with initial doses of the drug were considered

---

[5]"Branched-chain" logic refers to mechanisms by which portions of a decision network can be considered or ignored depending upon the data on a given case. For example, in an acid-base program the anion gap might be calculated and a branch-point could then determine whether the pathway for analyzing an elevated anion gap would be required. If the gap were not elevated, that whole portion of the logic network could be skipped.

in subsequent therapy recommendations [72], [89]. More recently, a third group in Boston, noting the insensitivity of the first two approaches to the kinds of nonnumeric observations that experts tend to use in modifying digitalis therapy, augmented the pharmacokinetic model with a patient-specific model of clinical status [31]. Running their system in a monitoring mode, in parallel with actual clinical practice on a cardiology service, they found that each patient in the trial in whom toxicity developed had received more digitalis than would have been recommended by their program.

4.2   Example

Perhaps the best known program in this category is the interactive system developed at Boston's Beth Israel Hospital by Bleich. Originally designed as a program for assessment of acid-base disorders [2], it was later expanded to consider electrolyte abnormalities as well [3], [4]. The knowledge in Bleich's program is a distillation of his own expertise regarding acid-base and electrolyte disorders. The system begins by collecting initial laboratory data from the physician seeking advice on a patient's management. Branched-chain logic is triggered by abnormalities in the initial data so that only the pertinent sections of the extensive decision pathways created by Bleich are explored. Essentially all questions asked by the program are numerical laboratory values or "yes-no" questions (e.g., "Does the patient have pitting edema?"). Depending upon the complexity and severity of the case, the program eventually generates an evaluation note that may vary in length from a few lines to several pages. Included are suggestions regarding possible causes of the observed abnormalities and suggestions for correcting them. Literature references are also provided.

Although the program was made available at several East Coast institutions, few physicians accepted it as an ongoing clinical tool. Bleich points out that part of the reason for this was the system's inherent educational impact; physicians simply began to anticipate its analysis after they had used it a few times [3]. More recently he has been experimenting with the program operating as a monitoring system[6], thereby avoiding direct interaction with the physician.

The system's lack of sustained acceptance by physicians is probably due to more than its educational impact, however. For example, there is no feedback in the system; every patient is seen as a new case and the program has no concept

---

[6]Personal communication with Dr. Bleich, 1975.

of following a patient's response to prior therapeutic measures. Furthermore, the program generates differential diagnosis lists but does not pursue specific etiologies; this can be particularly bothersome when there are multiple coexistent disturbances in a patient and the program simply suggests parallel lists of etiologies without noting or pursuing the possible interrelationships.

Finally, the system is highly individualized in that it contains consideration of specific relationships only when Bleich specifically thought to include them in the logic network. Of course human consultants also give personalized advice which may differ from that obtained from other experts. However, a group of researchers in Britain [79] who analyzed Bleich's program along with four other acid-base/electrolyte systems, found total agreement among the programs in only 20% of test cases when these systems were asked to define the acid-base disturbance and the degree of compensation present. Their analysis does not reveal which of the programs reached the correct decision, however, and it may be that the results are more an indictment of the other four programs than a valid criticism of the advice from Bleich's acid-base component.

4.3 Discussion of the Methodologies

The programs mentioned in this section are very different· in several respects, and each tends to overlap with other methodologies we have discussed. Bleich's program, for example, is essentially a complicated clinical algorithm interfaced with mathematical formulations of electrolyte and acid-base pathophysiology. As such it suffers from the weaknesses of all algorithmic approaches, most importantly its highly structured and inflexible logic which is unable to contend with unforseen circumstances not specifically included in the algorithm. The digitalis dosing programs all draw on mathematical techniques from the field of biomedical modeling (not discussed here), but have recently shown more reliance on methods from other areas as well. In particular these have included symbolic reasoning methods that allow clinical expertise to be captured and utilized in conjunction with mathematical techniques [31]. The Boston group that developed this most recent digitalis program is interested in similarly developing an acid-base/electrolyte system so that judgmental knowledge of experts can be interfaced with the mathematical models of pathophysiology[7].

_____

[7]Personal communication, 1978, with Prof. Peter Szolovits.

5   Statistical Pattern Matching Techniques

5.1   Overview

Pattern matching techniques define the mathematical relationship between measurable features and classifications of objects [12], [46]. In medicine, the presence or absence of each of several signs and symptoms in a patient may be definitive for the classification of the patient as "abnormal" or into the category of a specific disease. They are also used for prognosis [1], or predicting disease duration, time course, and outcomes. These techniques have been applied to a variety of medical domains, such as image processing and signal analysis, in addition to computer-assisted diagnosis.

In order to find the diagnostic pattern, or discriminant function, the method requires a training set of objects, for which the correct classification is already known, as well as reliable values for their measured features. If the form and parameters are not known for the statistical distributions underlying the features, then they must be estimated. Parametric techniques focus on learning the parameters of the probability density functions, while non-parametric (or "distribution-free") techniques make no assumptions about the form of the distributions. After training, then, the pattern can be matched to new, unclassified objects to aid in deciding the category to which the new object belongs[8].

There are numerous variations on this general methodology, most notably in the mathematical techniques used to extract characteristic measurements (the features) and to find and refine the pattern classifier during training. For example, linear regression analysis is a commonly used technique for finding the coefficients of an equation that defines a recurring pattern or category of diagnostic or prognostic interest. Recent work emphasizes structural relationships among sets of features more than statistical ones.

Three of the best known training criteria for the discriminant function are:

(a) Bayes' criterion: choose the function that has the minimum cost associated with incorrect diagnoses[9];

(b) clustering criterion: choose the function that produces the tightest clusters;

(c) least-squared-error criterion: choose the function that minimizes the squared differences between predicted and observed measurement values.

---

[8] It is possible to detect patterns, even without a known classification for objects in the training set, with so-called "unsupervised" learning techniques. Also, it is possible to work with both numerical and non-numerical measurements.

[9] See Section 6 for further discussion.

Ten commonly used mathematical models based on these criteria have been shown to produce remarkably similar diagnostic results for the same data [7].

5.2   Example

There are numerous papers reporting on the use of pattern recognition methods in medicine. Armitage [1] discusses three examples of prognostic studies, with an emphasis on regression methods. Siegel et al. [27] discuss uses of cluster analysis. One recent diagnostic application using Bayes' criterion [67] classifies patients having chest pains into three categories: D_1: acute myocardial infarction (MI); D_2: coronary insufficiency; and D_3: non-cardiac causes of chest pain. The need for early diagnosis of heart attacks without laboratory tests is a prevalent problem, yet physicians are known to misclassify about one third of the patients in categories D_1 and D_2 and about 80% of those in D_3. In order to determine the correct classification, each patient in the training set was classified after 3 days, based on laboratory data including electrocardiogram (ECG) and blood data (cardiac enzymes). There remained some uncertainty about several patients with "probable MI." Seventeen variables were selected from many: 9 features with continuous values (including age, heart rates, white blood count, and hemoglobin) and 8 features with discrete values (sex and 7 ECG features).

The training data were measurements on 247 patients. The decision rule was chosen using Bayes' theorem to compute the posterior probabilities of each diagnostic class given the feature vector X. $(X = [x_1, x_2, \ldots, x_{17}])$.[10]. Then a decision rule was chosen to minimize the probability of error, that is, to adjust the coefficients on the feature vector X [11] such that for the correct class D_i:

$$P(D_i|X) = \text{MAX} (P(D_1|X), P(D_2|X), P(D_3|X))$$

The class conditional probability density functions must be estimated initially, and the performance of the decision rule depends on the accuracy of the assumed model.

Using the same 247 patients for testing the approach, the trained

---

[10]The posterior probability of a diagnostic class, represented as $P(D_i|X)$, is the probability that a patient falls in diagnostic category D_i given that the feature vector X has been observed.

[11]See [56] for a study in which the coefficients are reported because of their medical import.

classifier averaged 80% correct diagnoses over the three classes, using only data available at the time of admission. Physicians, using more data than the computer, averaged only 50.5% correct over these three categories for the same patients. Training the classifier with a subset of the patients, and using the remainder for testing, produced nearly as good results.

5.3   Discussion of the Methodology

The number of reported medical applications of pattern recognition techniques is large, but there are also numerous problems associated with the methodology. The most obvious difficulties are choosing the set of features in the first place, collecting reliable measurements on a large sample, and verifying the initial classifications among the training data. Current techniques are inadequate for problems in which trends or movement of features are important characteristics of the categories. Also the problems for which existing techniques are accurate are those that are well characterized by a small number of features ("dimensions of the space").

As with all techniques based on statistics, the size of the sample used to define the categories is an important consideration. As the number of important features and the number of relevant categories increase, the required size of the training set also increases. In one test [7], pattern classifiers trained to discriminate among 20 disease categories from 50 symptoms were correct 51% - 64% of the time. The same methods were used to train classifiers to discriminate between 2 of the diseases, from the same 50 symptoms, and produced correct diagnoses 92% - 98% of the time.

The context in which a local pattern is identified raises problems related to the issue of utilizing medical knowledge. It is difficult to find and use classifiers that are best for a small decision, such as whether an area of an X-ray is inside or outside the heart, and integrate those into a global classifier, such as one for abnormal heart volume.

Accurate application of a classifier in a hospital setting also requires that the measurements in that clinical environment are consistent with the measurements used to train the classifier initially. For example, if diseases and symptoms are defined differently in the new setting, or if lab test values are reported in different ranges — or different lab tests used — then decisions based on the classification are not reliable.

Pattern recognition techniques are often misapplied in medical domains in

which the assumptions are violated. Some of the difficulties noted above are avoided in systems that integrate structural knowledge into the numerical methods and in systems that integrate human and machine capabilities into single, interactive systems. These modifications will overcome one of the major difficulties seen in completely automated systems, that of providing the system with good "intuitions" based on an expert's a priori knowledge and experience [46].

6   Bayesian Statistical Approaches

6.1   Overview

More work has been done on Bayesian approaches to computer-based medical decision making than on any of the other methodologies we have discussed. The appeal of Bayes' Theorem [12] is clear: it potentially offers an exact method for computing the probability of a disease based on observations and data regarding the frequency with which these observations are known to occur for specified diseases. In several domains the technique has been shown to be exceedingly accurate, but there are also several limitations to the approach which we discuss below.

In its simplest formulation, Bayes' Theorem can be seen as a mechanism to calculate the probability of a disease, in light of specified evidence, from the a priori probability of the disease and the conditional probabilities relating the observations to the diseases in which they may occur. For example, suppose disease D_i is one of n mutually exclusive diagnoses under consideration and E is the evidence or observations supporting that diagnosis. Then if $P(D_i)$ is the a priori probability of the ith disease:

$$P(D_i|E) = \frac{P(D_i) \; P(E|D_i)}{\sum_{j=1}^{n} P(D_j) \; P(E|D_j)}$$

The theorem can also be represented or derived in a variety of other forms, including an odds/likelihood ratio formulation. We cannot include such details here, but any introductory statistics book or Lusted's classic volume [58] presents the subject in considerable detail.

---

[12]also often referred to as Bayes' rule, discriminant, or criterion

Among the most commonly recognized problems with the utilization of a Bayesian approach is the large amount of data required to determine all the conditional probabilities needed in the rigorous application of the formula. Chart review or computer-based analysis of large databanks occasionally allows most of the necessary conditional probabilities to be obtained. A variety of additional assumptions must be made. For example: (1) the diseases under consideration are assumed mutually exclusive and exhaustive (i.e., the patient is assumed to have one of the n diseases, (2) the clinical observations are assumed to be conditionally independent over a given disease[13], and (3) the incidence of the symptoms of a disease is assumed to be stationary (i.e., the model generally does not allow for changes in disease patterns over time).

One of the earliest Bayesian programs was Warner's system for the diagnosis of congenital heart disease [107]. He compiled data on 83 patients and generated a symptom-disease matrix consisting of 53 symptoms (attributes) and 35 disease entities. The diagnostic performance of the computer, based on the presence or absence of the 53 symptoms in a new patient, was then compared to that of two experienced physicians. The program was shown to reach diagnoses with an accuracy equal to that of the experts. Furthermore, system performance was shown to improve as the statistics in the symptom-disease matrix stabilized with the addition of increasing numbers of patients.

In 1968 Gorry and Barnett pointed out that Warner's program had required making all 53 observations for every patient to be diagnosed, a situation which would not be realistic for many clinical applications. They therefore utilized a modification of Bayes' Theorem in which observations are considered sequentially. Their computer program analyzed observations one at a time, suggested which test would be most useful if performed next, and included termination criteria so that a diagnosis could be reached, when appropriate, without needing to make all the observations [28]. Decisions regarding tests and termination were made on the basis of calculations of expected costs and benefits at each step in the logical process[14]. Using the same symptom-disease matrix developed by Warner, they were able to attain equivalent diagnostic

---

[13]The purest form of Bayes' Theorem allows conditional dependencies, and the order in which evidence is obtained, to be explicitly considered in the analysis. However, the number of required conditional probabilities is so unwieldy that conditional independence of observations, and non-dependence on the order of observations, is generally assumed [101].

[14]See the decision theory discussion in Section 7.

performance using only 6.9 tests on average[15]. They pointed out that, because the costs of medical tests may be significant (in terms of patient discomfort, time expended, and financial expense), the use of inefficient testing sequences should be regarded as ineffective diagnosis. Warner has also more recently included Gorry and Barnett's sequential diagnosis approach in an application regarding structured patient history-taking [110].

The medical computing literature now includes many examples of Bayesian diagnosis programs, most of which have used the nonsequential approach, in addition to the necessary assumptions of symptom independence and mutual exclusivity of disease as discussed above. One particularly successful research effort has been chosen for discussion.

6.2   Example

Since the late 1960's deDombal and associates, at the University of Leeds in England, have been studying the diagnostic process and developing computer-based decision aids using Bayesian probability theory. Their area of investigation has been gastrointestinal diseases, originally acute abdominal pain [10] with more recent analyses of dyspepsia [39] and gastric carcinoma [125].

Their program for assessment of acute abdominal pain was evaluated in the emergency room of their affiliated hospital [10]. Emergency physicians filled out data sheets summarizing clinical and laboratory findings on 304 patients presenting with abdominal pain of acute onset. The data from these sheets became the attributes that were subjected to Bayesian analysis; the required conditional probabilities had been previously compiled from a large group of patients with one of 7 possible diagnoses[16]. Thus the Bayesian formulation assumed each patient had one of these diseases and would select the most likely on the basis of recorded observations. Diagnostic suggestions were obtained in batch mode and did not require direct interaction between physician and computer; the program could generate results in from 30 seconds to 15 minutes depending upon the level of system use at the time of analysis [38]. Thus the computer output could have been made available to the emergency room physician, on average, within 5 minutes after the data form was completed and handed to the technician assisting with the study.

---

[15]Tests for determining attributes were defined somewhat differently than they had been by Warner. Thus the maximum number of tests was 31 rather than the 53 observations used in the original study.

[16]appendicitis, diverticulitis, perforated ulcer, cholecystitis, small bowel obstruction, pancreatitis, and non-specific abdominal pain.

During the study [10], however, these computer-generated diagnoses were simply saved and later compared to (a) the diagnoses reached by the attending clinicians, and (b) the ultimate diagnosis verified at surgery or through appropriate tests. Although the clinicians reached the correct diagnosis in only 65%-80% of the 304 cases (with accuracy depending upon the individual's training and experience), the program was correct in 91.8% of cases. Furthermore, in 6 of the 7 disease categories the computer was proved more likely than the senior clinician in charge of a case to assign the patient to the correct disease category. Of particular interest was the program's accuracy regarding appendicitis - a diagnosis which is often made incorrectly. In no cases of appendicitis did the computer fail to make the correct diagnosis, and in only six cases were patients with non-specific abdominal pain incorrectly classified as having appendicitis. Based on the actual clinical decisions, however, over 20 patients with non-specific abdominal pain were unnecessarily taken to surgery for appendicitis, and in six cases patients with appendicitis were "watched" for over eight hours before they were finally taken to the operating room.

These investigators also performed a fascinating experiment in which they compared the program's performance based on data derived from 600 real patients, with the accuracy the system achieved using "estimates" of conditional probabilities obtained from experts [54][17]. As discussed above, the program was significantly more effective than the unaided clinician when real-life data were utilized. However, it performed significantly _less_ well than clinicians when expert estimates were used. The results supported what several other observers have found, namely that physicians often have very little idea of the "true" probabilities for symptom-disease relationships.

Another Leeds study of note was an analysis of the effect of the system on the performance of clinicians [11]. The trial we have mentioned that involved 304 patients was eventually extended to 552 before termination. Although the computer's accuracy remained in the range of 91% throughout this period, the performance of clinicians was noted to improve markedly over time. Fewer negative laparotomies were performed, for example, and the number of acute appendices that perforated (ruptured) also declined. However, these data reverted to baseline after the study was terminated, suggesting that the

---

[17]Such estimates are referred to as "subjective" or "personal" probabilities, and some investigators have argued that they should be utilized in Bayesian systems when formally derived conditional probabilities are not available [58].

constant awareness of computer monitoring and feedback regarding system performance had temporarily generated a heightened awareness of intellectual processes among the hospital's surgeons.

6.3   Discussion of the Methodology

The ideal matching of the problem of acute abdominal pain and Bayesian analysis must also be emphasized; the methodology cannot necessarily be as effectively applied in other medical domains where the following limitations of the Bayesian approach may have a greater impact.

(1) The assumption of conditional independence of symptoms usually does not apply and can lead to substantial errors in certain settings [66]. This has led some investigators to seek new numerical techniques that avoid the independence assumption [8]. If a pure Bayesian formulation is utilized without making the independence assumption, however, the number of required conditional probabilities becomes prohibitive for complex real world problems [101].

(2) The assumption of mutual exclusivity and exhaustiveness of disease categories is usually false. In actual practice concurrent and overlapping disease categories are common. In deDombal's system, for example, many of .the abdominal pain diagnoses missed were outside the seven "recognized" possibilities; if a program starts with an assumption that it need only consider a small number of defined likely diagnoses, it will inevitably miss the rare or unexpected cases - precisely the ones with which the clinician is most apt to need assistance.

(3) In many domains it may be inaccurate to assume that relevant conditional probabilities are stable over time (e.g., the likelihood that a particular bacterium will be sensitive to a specific antibiotic). Furthermore, diagnostic categories and definitions are constantly changing, as are physicians' observational techniques, thereby invalidating data previously accumulated. A similar problem results from variations in a priori probabilities depending upon the population from which a patient is drawn. Some observers feel that these are major limitations to the use of Bayesian techniques [13].

In general, then, a purely Bayesian approach can so constrain problem formulation as to make a particular application unrealistic and hence unworkable. Furthermore, even when diagnostic performance is excellent such as in deDombal's approach to abdominal pain evaluation, clinical implementation and system acceptance will generally be difficult.

7   Decision Theoretical Approaches

7.1   Overview

Bayes' Theorem is only one of several techniques used in the larger field of decision analysis, and there has recently been increasing interest in the ways in which decision theory might be applied to medicine and adapted for automation. Several excellent reviews of the field are available in basic reviews [40], textbooks [78], and medically-oriented journal articles [61], [87], [102]. In general terms, decision analysis can be seen as any attempt to consider _values_ associated with choices, as well as probabilities, in order to analyze the processes by which decisions are made or should be made. Schwartz identifies the calculation of "expected value" as central to formal decision analysis [87]. Ginsberg contrasts medical classification problems (e.g., diagnosis) with broader decision problems (e.g., "What should I do for this patient?"), and asserts that most important medical decisions fall in the latter category and are best approached through decision analysis [25]. The following topics are among the central issues in the field.

(1) _Decision Trees_. The decision making process can be seen as a sequence of steps in which the clinician selects a path through a network of plausible events and actions. Nodes in this tree-shaped network are of two kinds: _decision nodes_, where the clinician must choose from a set of actions, and _chance nodes_, where the outcome is not directly controlled by the clinician but is a probabilistic response of the patient to some action taken. For example, a physician may choose to perform a certain test (decision node) but the occurrence or nonoccurrence of complications may be largely a matter of statistical likelihood (chance node). By analyzing a difficult decision process before taking any actions, it may be possible to delineate in advance all pertinent chance and decision nodes, all plausible outcomes, plus the paths by which these outcomes might be reached. Furthermore, data may exist to allow specific probabilities to be associated with each chance node in the tree.

(2) _Expected Values_. In actual practice physicians make sequential decisions based on more than the probabilities associated with the chance node that follows. For example, the best possible outcome is not necessarily sought if the costs associated with that "path" far outweigh those along alternate pathways (e.g., a definitive diagnosis may not be sought if the required testing procedure is expensive or painful and patient management will be unaffected;

similarly, some patients prefer to "live with" an inquinal hernia rather than undergo a surgical repair procedure). Thus anticipated "costs" (financial, complications, discomfort, patient preference) can be associated with the decision nodes. Using the probabilities at chance nodes, the costs at decision nodes, and the "value" of the various outcomes, an "expected value" for each pathway through the tree (and in turn each node) can be calculated. The ideal pathway, then, is the one which maximizes the expected value.

(3) Eliciting Values. Obtaining from physicians and patients the cost and values they associate with various tests and outcomes can be a formidable problem, particularly since formal analysis requires expressing the various costs in standardized units. One approach has been simply to ask for value ratings on a hypothetical scale, but it can be difficult to get the physician or patient to keep the values[18] separate from their knowledge of the probabilities linked to the associated chance nodes. An alternate approach has been the development of lottery games. Inferences regarding values can be made by identifying the odds, in a hypothetical lottery, at which the physician or patient is indifferent regarding taking a course of action with certain outcome and betting on a course with preferable outcome but with a finite chance of significant negative costs if the "bet" is lost. In certain settings this approach may· be accepted and provide important guidelines in decision making [71].

(4) Test Evaluation. Since the tests which lie at decision nodes are central to clinical decision analysis, it is crucial to know the predictive value of tests that are available. This leads to consideration of test sensitivity, specificity, receiver operator characteristic curves, and sensitivity analysis. Such issues are discussed by Komaroff et al. in this issue of the PROCEEDINGS and have also been summarized elsewhere in the clinical literature [62].

Many of the major studies of clinical decision analysis have not specifically involved computer implementations. Schwartz et al. examined the workup of renal vascular hypertension, developing arguments to show that for certain kinds of cases a purely qualitative theoretical approach was feasible and useful [87]. However, they showed that for more complex clinically challenging cases the decisions could not be adequately sorted out without the introduction of numerical techniques. Since it was impractical to assume that

---

[18]also termed "utilities" in some references; hence the term "utility theory" [78].

clinicians would ever take the time to carry out a detailed quantitative decision analysis by hand, they pointed out the logical role for the computer in assisting with such tasks and accordingly developed the system we discuss as an example below [29].

Other colleagues of Schwartz at Tufts have been similarly active in applying decision theory to clinical problems. Pauker and Kassirer have examined applications of formal cost-benefit analysis to therapy selection [68] and Pauker has also looked at possible applications of the theory to the management of patients with coronary artery disease [70]. An entire issue of the New England Journal of Medicine has also been devoted to papers on this methodology [41].

7.2   Example

Computer implementations of clinical decision analysis have appeared with increasing frequency since the mid-1960's. Perhaps the earliest major work was that of Ginsberg at Rand Corporation [24], with more recent systems reported by Pliskin and Beck [74] and Safran et al. [85].

We will briefly describe here the program of Gorry et al., developed for the management of acute renal failure [29]. Drawing upon Gorry's experience with the sequential Bayesian approach previously mentioned [28], the investigators recognized the need to incorporate some way of balancing the dangers and discomforts of a procedure against the value of the information to be gained. They divided their program into two parts: phase I considered only tests with minimal risk (e.g., history, examination, blood tests) and phase II considered procedures involving more risk and inconvenience. The phase I program considered 14 of the most common causes of renal failure and utilized a sequential test selection process based on Bayes' Theorem and omitting more advanced decision theoretical methodology [28]. The conditional probabilities utilized were subjective estimates obtained from an expert nephrologist and were therefore potentially as problematic as those discussed by Leaper et al. [54] (see Section 6.2). The researchers found that they had no choice but to use expert estimates, however, since detailed quantitative data were not available either in databanks nor the literature.

It is in the phase II program that the methods of decision theory were employed because it was in this portion of the decision process that the risks of procedures became important considerations. At each step in the decision

process this program considers whether it is best to treat the patient immediately or to first carry out an additional diagnostic test. To make this decision the program identifies the treatment with the highest current expected value (in the absence of further testing), and compares this with the expected values of treatments that could be instituted if another diagnostic test were performed. Comparison of the expected values are made in light of the risk of the test in order to determine whether the overall expected value of the test is greater than that of immediate treatment. The relevant values and probabilities of outcomes of treatment were obtained as subjective estimates from nephrologists in the same way that symptom-disease data had been obtained. All estimates were gradually refined as they gained experience using the program, however.

The program was evaluated on 18 test cases in which the true diagnosis was uncertain but two expert nephrologists were willing to make management decisions. In 14 of the cases the program selected the same therapeutic plan or diagnostic test as was chosen by the experts. For three of the four remaining cases the program's decision was the physicians' second choice and was, they felt, a reasonable alternative plan of action. In the last case the physicians also accepted the program's decision as reasonable although it was not among their first two choices.

7.3 Discussion of the Methodology

The excellent performance of Gorry's program, despite its reliance on subjective estimates from experts, may serve to emphasize the importance of the clinical analysis that underlies the decision theoretical approach. The reasoning steps in managing clinical cases have been dissected in such detail that small errors in the probability estimates are apparently much less important than they were for deDombal's purely Bayesian approach [54]. Gorry suggests this may be simply because the decisions made by the program are based on the combination of large aggregates of such numbers, but this argument should apply equally for a Bayesian system. It seems to us more likely that distillation of the clinical domain in a formal decision tree gives the program so much more knowledge of the clinical problem that the quantitative details become somewhat less critical to overall system operation. The explicit decision network is a powerful knowledge structure; the "knowledge" in deDombal's system lies in conditional probabilities alone and there is no larger

scheme to override the propagation of error as these probabilities are mathematically manipulated by the Bayesian routines.

The decision theory approach is not without problems, however. Perhaps the most difficult problem is assigning numerical values (e.g., dollars) to a human life or a day of health, etc. Some critics feel this is a major limitation to the methodology [112]. Overlapping or coincident diseases are also not well-managed, unless specifically included in the analysis, and the Bayesian foundation for many of the calculations still assumes mutually exclusive and exhaustive disease categories. Problems of symptom conditional dependence still remain, and there is no easy way to include knowledge regarding the time course of diseases. Gorry points out that his program was also incapable of recognizing circumstances in which two or more actions should be carried out concurrently. Furthermore decision theory per se does not provide the kind of focusing mechanisms that clinicians tend to use when they assume an initial diagnostic hypothesis in dealing with a patient and discard it only if subsequent data make that hypothesis no longer tenable. Other similar strategies of clinical reasoning are becoming increasingly well-recognized [48] and account in large part for the applications of symbolic reasoning techniques to be discussed in the next section.

8   Symbolic Reasoning Approaches

8.1   Overview

In the early 1970's researchers at several institutions simultaneously began to investigate the potential applications to clinical decision making of symbolic reasoning techniques drawn from the branch of computer science known as artificial intelligence (AI). The field is well-reviewed in a recent book by Winston [120]. Although the term "artificial intelligence" has never been uniformly defined, it is generally accepted to include those computer applications in which the tasks require largely symbolic inference rather than numeric calculation. Examples include programs that reason about mineral exploration, organic chemistry, or molecular biology; programs that converse in English and understand spoken sentences; and programs that generate theories from observations.

Such programs gain their power from qualitative, experimental judgments – codified in so-called "rules-of-thumb" or "heuristics" – in contrast to

numerical calculation programs whose power derives from the analytical equations used. The heuristics focus the attention of the reasoning program on parts of the problem that seem most critical and parts of the knowledge base that seem most relevant. They also guide the application of the domain knowledge to an individual case by deleting items from consideration as well as focusing on items. The result is that these programs pursue a line of reasoning as opposed to following a sequence of steps in a calculation. Among the earliest symbolic inference programs in medicine was the diagnostic interviewing system of Kleinmuntz [49]. Other early work included Wortman's information processing system, the performance of which was largely motivated by a desire to understand and simulate the psychological processes of neurologists reaching diagnoses [121].

It was a landmark paper by Gorry in 1973, however, that first critically analyzed conventional approaches to computer-based clinical decision making and outlined his motivation for turning to newer symbolic techniques [30]. He used the acute renal failure program discussed in Section 7.2 [29] as an example of the problems arising when decision analysis is used alone. In particular, he analyzed some of the cases on which the renal failure program had failed but the physicians considering the cases had performed well. His conclusions from these observations include the following four points.

(1) Clinical judgment is based less on detailed knowledge of pathophysiology than it is on gross chunks of knowledge and a good deal of detailed experience from which rules of thumb are derived.

(2) Clinicians know facts, of course, but their knowledge is also largely judgmental. The rules they learn allow them to focus attention and generate hypotheses quickly. Such heuristics permit them to avoid detailed search through the entire problem space.

(3) Clinicians recognize levels of belief or certainty associated with many of the rules they use, but they do not routinely quantitate or utilize these certainty concepts in any formal statistical manner.

(4) It is easier for experts to state their rules in response to perceived misconceptions in others than it is for them to generate such decision criteria a priori.

In the renal failure program medical knowledge had been embedded in the structure of the decision tree. This knowledge was never explicit, and additions to the experts' judgmental rules had generally required changes to the tree itself.

Based on observations such as those above, Corry identified at least three important problems for investigation:

(1) Concept Formation. Clinical decision aids had traditionally had no true "understanding" of medicine. Although explicit decision trees had given the decision theory programs a greater sense of the pertinent associations, medical knowledge and the heuristics for problem solving in the field had never been explicitly represented nor utilized. So-called "common sense" was often clearly lacking when the programs failed, and this was often what most alienated potential physician users.

(2) Language Development. Both for capturing knowledge from collaborating experts, and for communicating with physician users, Gorry argued that further research on the development of computer-based linguistic capabilities was crucial.

(3) Explanation. Diagnostic programs had seldom emphasized an ability to explain the basis for their decisions in terms understandable to the physician. System acceptability was therefore inevitably limited; the physician would often have no basis for deciding whether to accept the program's advice, and might therefore resent what could be perceived as an attempt to dictate the practice of medicine.

Gorry's group at MIT and Tufts developed new approaches to examining the renal failure problem in light of these observations [69].

Due to the limitations of the older techniques, it was perhaps inevitable that some medical researchers would turn to the AI field for new methodologies. Major research areas in AI include knowledge representation, heuristic search, natural language understanding and generation, and models of thought processes — all topics clearly pertinent to the problems we have been discussing. Furthermore, AI researchers were beginning to look for applications to which they could apply some of the techniques they had developed in theoretical domains. This community of researchers has grown in recent years, and a recent issue of Artificial Intelligence was devoted entirely to applications of AI to biology, medicine and chemistry [98][19].

---

[19]Many of the systems described in this issue were developed on the SUMEX-AIM computing resource, a nationally shared system devoted entirely to applications of AI to the biomedical sciences. The SUMEX-AIM computer is physically located at Stanford University but is used by researchers nationwide via connections to the TYMNET. The resource is funded by the Division of Research Resources, Biotechnology Branch, National Institutes of Health.

Among.the programs using symbolic reasoning techniques are several systems that have been particularly novel and successful. Pople and Myers have developed a system called INTERNIST that assists with test selection for the diagnosis of all diseases in internal medicine [75]. This awesome task has been remarkably successful to date, with the program correctly diagnosing a large percentage of the complex cases selected from clinical pathologic conferences in the major medical journals[20]. The program utilizes a hierarchic disease categorization, an ad hoc scoring system for quantifying symptom-disease relationships, plus some clever heuristics for focusing attention, discriminating between competing hypotheses, and diagnosing concurrent diseases [76]. The system currently has an inadequate human interface, however, and is not yet implemented for clinical trials.

At Rutgers University Weiss', Kulikowski, and Safir have developed a model of ophthomologic reasoning regarding disease processes in the eye, specifically glaucoma [117]. In this specialized application area it has been possible to map relationships between observations, pathophysiologic states, and disease categories. The resulting causal associational network (termed CASNET) forms the basis for a reasoning program that gives advice regarding disease states in glaucoma patients and generates management recommendations.

For the AI researchers the question of how best to manage uncertainty in medical reasoning remains a central issue. All the programs mentioned have developed ad hoc weighting programs and avoided formal statistical approaches. Others have turned to the work of statisticians and philosophers of science who have devised theories of approximate or inexact reasoning. For example, Wechsler [114] describes a program that is based upon Zadeh's fuzzy set theory [124]. Shortliffe and Buchanan [94] have turned to confirmation theory for their model of inexact reasoning in medicine.

8.2   Example

The symbolic reasoning program selected for discussion is the MYCIN System at Stanford University [95]. The researchers cited a variety of design considerations which motivated the selection of AI methodologies for the consultation system they were developing [92]. They primarily wanted it to be useful to physicians and therefore emphasized the selection of a problem domain in which physicians had been shown to err frequently, namely the selection of

---

[20]Data communicated by Drs. Pople and Myers at the Second Annual A.I.M. Workshop, Rutgers University, June 1976.

antibiotics for patients with infections. They also cited human issues that they felt were crucial to make the system acceptable to physicians:

(1) it should be able to explain its decisions in terms a line of reasoning that a physician can understand;

(2) it should be able to justify its performance by responding to questions expressed in simple English;

(3) it should be able to "learn" new information rapidly by interacting directly with experts;

(4) its knowledge should be easily modifiable so that perceived errors can be corrected rapidly before they recur in another case; and

(5) the interaction should be engineered with the user in mind (in terms of prompts, answers, and information volunteered by the system as well as by the users).

All these design goals were based on the observation that previous computer decision aids had generally been poorly accepted by physicians, even when they were shown to perform well on the tasks for which they were designed. MYCIN's developers felt that barriers to acceptance were largely conceptual and could be counteracted in large part if a system were perceived as a clinical <u>tool</u> rather than a dogmatic replacement for the primary physician's own reasoning.

Knowledge of infectious diseases is represented in MYCIN as production rules, each containing a "packet" of knowledge obtained from collaborating experts [95][21]. A production rule is simply a conditional statement which relates observations to associated inferences that may be drawn. For example, a MYCIN rule might state that "<u>if</u> a bacterium is a gram positive coccus growing in chains, <u>then</u> it is apt to be a streptococcus." MYCIN's power is derived from such rules in a variety of ways:

(1) it is the program that determines which rules to use and how they should be chained together to make decisions about a specific case[22];

(2) the rules can be stored in a machine-readable format but translated into English for display to physicians;

(3) by removing, altering, or adding rules, the system's knowledge structures can be rapidly modified without explicitly restructuring the entire knowledge base; and

(4) the rules themselves can often form a coherent explanation of system reasoning if the relevant ones are translated into English and displayed in response to a user's question.

Associated with all rules and inferences are numerical weights reflecting the degree of certainty associated with them. These numbers, termed certainty factors, form the basis for the system's inexact reasoning in this complex task

---

[21]Production rules are a methodology frequently employed in AI research [9] and effectively applied to other scientific problem domains [6].

[22]The control structure utilized is termed "goal-oriented" and is similar to the consequent-theorem methodology used in Hewitt's PLANNER [37].

domain [94]. They allow the judgmental knowledge of experts to be captured in rule form and then utilized in a consistent fashion.

The MYCIN System has been evaluated regarding its performance at therapy selection for patients with either septicemia [123] or meningitis [122]. The program performs comparably with experts in these two task domains, but as yet it has no rules regarding the other infectious disease problem areas. Further knowledge base development will therefore be required before MYCIN is made available for clinical use; hence questions regarding its acceptability to physicians cannot yet be assessed. However, the required implementation stages have been delineated [93], attention has been paid to all the design criteria mentioned above, and the program does have a powerful explanation capability [88].

8.3 Discussion of the Methodology

Symbolic reasoning techniques differ from the other methodologies mentioned in this article in that the computer techniques themselves are as yet experimental and rapidly changing. Whereas the computations involved in Bayes' Theorem, for example, involve straightforward application of computing techniques already well-developed, basic researchers in computer science continue to develop new methodologies for knowledge representation, language understanding, heuristic search, and the other symbolic reasoning problems we have mentioned. Thus the AI programs tend to be developed in highly experimental environments where short term practical results are often unlikely to be found. The programs typically require large amounts of space and tend to be slow, particularly in time-sharing environments. As has been true for most of the methodologies discussed, AI researchers have still not developed adequate methods for handling concurrent diseases, assessing the time course of disease, nor acquiring adequate structured knowledge from experts. Furthermore, inexact reasoning techniques tend to be developed and justified largely on intuitive grounds.

Despite these significant limitations, the techniques of artificial intelligence do provide a way to respond to many of Gorry's observations regarding the inadequacies of prior methodologies as described above [30]. There are now several programs responsive to his criticisms. Szolovits and Pauker have recently reviewed some applications of AI to medicine and have attempted to weigh the successes of this young field against the very real

problems that lie ahead [101]. They identify several serious deficiencies of current systems. For example, termination criteria are still poorly understood. Although INTERNIST can diagnose simultaneous diseases, it also pursues all abnormal findings to completion, even though a clinician often ignores minor unexplained abnormalities if the rest of a patient's clinical status is well understood. In addition, although some of these programs now cleverly mimic some of the reasoning styles observed in experts [14],[48], it is less clear how to keep the systems from abandoning one hypothesis and turning to another one as soon as new information suggests another possibility. Programs that operate this way appear to digress from one topic to another — a characteristic that decidedly alienates a user regardless of the validity of the final diagnosis or advice.

9   Conclusions

This review has shown that there are two recurring issues to confront in considering the field of computer-based clinical decision making:

(1) How can we design systems that reach better, more reliable decisions in a broad range of applications, and

(2) How can we more effectively encourage the use of such systems by physicians or other intended users?

We shall summarize by reviewing these points separately.

Performance Issues

Central to assuring a program's adequate performance is a matching of the most appropriate technique with the problem domain. We have seen that the structured logic of clinical algorithms can be effectively applied to triage functions and other primary care problems, but they would be less naturally matched with complex tasks such as the diagnosis and management of acute renal failure. Good statistical data may support an effective Bayesian program in settings where diagnostic categories are small in number, non-overlapping, and well-defined, but the lack of higher level domain knowledge limits the effectiveness of the Bayesian approach in more complex patient management or diagnostic environments. A mathematical approach may support decision making in certain well-described fields in which observations are typically quantified, and related by functional expressions. These examples, and others, demonstrate the the need for thoughtful consideration of the technique most appropriate for managing a clinical problem. In general the simplest effective methodology is

to be preferred, but acceptability issues must also be considered as discussed below.

It is also always appropriate to ask whether computer-based approaches are needed at all for a given decision making task. The clinical algorithm developers, for example, have almost uniformly discarded the machine, and Schwartz et al. pointed out that a useful decision analysis can often be accomplished in a qualitative manner using paper and pencil [87].

Finally, it is important to consider the extent to which a program's "understanding" of its task domain will heighten its performance, particularly in settings where knowledge of the field tends to be highly judgmental and poorly quantified. We use the term "understanding" here to refer to the degree of judgmental or structural knowledge (as opposed to data) that is contained in the program. Analyses of human clinical decision making [14], [48] suggest that as decisions move from simple to complex, a physician's reasoning style becomes less algorithmic and more heuristic, with qualitative judgmental knowledge and the conditions for invoking it coming increasingly into play. It is likely that medical computing researchers will similarly have to become "knowledge engineers" in the sense that they will look for effective ways to match the knowledge structures that they use to the complexity of the tasks they are undertaking.

Acceptability Issues

A recurring observation as one reviews the literature of computer-based medical decision making is that essentially none of the systems has been effectively utilized outside of a research environment, _even when its performance has been shown to be excellent_! This suggests that it may be an error to concentrate our research effort primarily on improving the decision making performance of computers when there is evidently much more required before these systems will have clinical impact. It is tempting to conclude that the biases of medical personnel against computers are so strong that systems will inevitably be rejected, regardless of performance, and in fact there are some data to support this view [99]. However, we are beginning to see examples of applications in which initial resistance to automated techniques has gradually been overcome through the incorporation of adequate system benefits [113].

Perhaps one of the most revealing lessons on this subject is an observation

regarding the system of Mesel et al. that we described earlier [64]. Despite documented physician resistance to clinical algorithms in other settings [34], the physicians in Mesel's study accepted the guidance of protocols for the management of chemotherapy in their cancer patients. It is likely that the key to acceptance in this instance is the fact that these physicians had previously had no choice but to refer their patients with cancer to the tertiary care center in Birmingham where all complex chemotherapy was administered. The introduction of the protocols permitted these physicians to undertake tasks that they had previously been unable to do, and it simultaneously allowed maintenance of close doctor- patient relationships and helped the patients avoid frequent long trips to the center. The motivation for the physician to use the system is clear in this case. It is reminiscent of Rosati's assertion that physicians will first welcome computer decision aids when they become aware that colleagues who are using the machine have a clear advantage in their practice [81].

A heightened awareness of "human engineering" issues among medical computing researchers is also apt to help improve acceptance of computers by physicians. Fox has recently reviewed this field in detail [18]. The issues range from the mechanics of interaction at a computer terminal to program characteristics designed to make the system appear as a tool for the physician rather than a dogmatic advice-giving machine.

Adequate attention must also be given to the severe time constraints perceived by physicians. Ideally they would like programs to take no more time than they currently spend when accomplishing the same task on their own. Time and schedule pressures are similarly likely to explain the greater resistance to automation among interns and residents than among medical students or practicing physicians in Startsman's study [99].

Finally it must be noted that acceptability issues should generally be considered from the outset in a system's design because they may dictate the choice of methodology as much as the task domain itself does. The role of formal knowledge structures to facilitate explanation capabilities, for example, may argue in favor of using symbolic reasoning techniques even when a somewhat less complex methodology might have been adequate for the decision task.

In summary, the trend towards increased use of knowledge engineering techniques for clinical decision programs has been in response to desires for both improved performance and improved acceptance of such systems. As greater

experience is gained with these techniques and they become better known throughout the medical computing community, it is likely that we will see increasingly powerful unions between symbolic reasoning and the alternate methodologies we have discussed. One lesson to be drawn lies in the recognition that there is basic computer science research to be done in medical computing, and that the field is more than the application of established computing techniques in medical domains.

Acknowledgments

References

1. Armitage, P. and Gehan, E.A. "Statistical methods for the identification and use of prognostic factors." _Int. J. Cancer_, 13, pp. 16-36, (1974).

2. Bleich, H.L. "Computer evaluation of acid-base disorders." _J. Clin. Invest._ 48, pp. 1689-1696 (1969).

3. Bleich, H.L. "The computer as a consultant." _N. Eng. J. Med._ 284, pp. 141-147 (1971).

4. Bleich, H.L. "Computer-based consultation: electrolyte and acid-base disorders." _Amer. J. Med_ 53, pp. 285-291 (1972).

5. Blum, R.L. and Wiederhold, G. " Inferring knowledge from clinical data banks: utilizing techniques from artificial intelligence," _Proc. 2nd Ann. Symp. on Comp. Applic. in Med. Care_, IEEE, Washington D.C., November 1978, pp. 303-307.

6. Buchanan, B.G. and Feigenbaum, E.A. "Dendral and Meta-Dendral: their applications dimension." _Artificial Intelligence_ 11, pp. 5-24 (1978).

7. Croft, D.J. "Is computerized diagnosis possible?" _Comp. Biomed. Res._ 5, pp. 351-367 (1972).

8. Cumberpatch, J. and Heaps, H.S. "A disease-conscious method for sequential diagnosis by use of disease probabilities without assumption of symptom independence." _Int. J. Biomed. Comput._ 7, pp. 61-78 (1976).

9. Davis, R. and King, J. "An overview of production systems." In _Machine Representation of Knowledge_ (E.W. Elcock and D. Michie, eds.), New York: Wiley, 1976.

10. deDombal, F.T., Leaper, D.J., Staniland, J.R., et al. "Computer-aided diagnosis of acute abdominal pain." _Brit. Med. J._ 2, pp.9-13 (1972).

11. deDombal, F.T., Leaper, D.J., Horrocks, J.C., et al. "Human and computer-aided diagnosis of abdominal pain: further report with emphasis on performance of clinicians." _Brit. Med. J._ 1, pp.376-380 (1974).

12. Duda, R.O. and Hart, P.E. _Pattern Classification and Scene Analysis._ New York: Wiley, 1973.

13. Edwards, W. "N=1: diagnosis in unique cases." In _Computer Diagnosis And Diagnostic Methods_, (J.A. Jacquez, ed.), Springfield, Ill.: Charles C. Thomas, 1972, pp. 139-151.

14. Elstein, A.S., Shulman, L.S., and Sprafka, S.A. _Medical Problem Solving: An Analysis of Clinical Reasoning._ Cambridge, Mass.: Harvard Univ. Press, 1978.

15. Feigenbaum, E.A. "The Art of Artificial Intelligence: Themes and case studies of knowledge engineering." _AFIPS Conference Proc., NCC 1978._ Vol. 47. Montvale, N.J.: AFIPS Press, 1978, p.227.

16. Feinstein, A.R. "Quality of data in the medical record." _Comput. Biomed. Res._ 3, pp. 426-435 (1970).

17. Feinstein, A.R., Rubinstein, J.F., and Ramshaw, W.A. "Estimating prognosis with the aid of a conversational mode computer program." _Anns. Int. Med._ 76, pp. 911-921 (1972).

18. Fox, J. "Medical computing and the user." _Int. J. Man-Machine Studies_ 9, pp. 669-686 (1977).

19. Friedman, R.B. and Gustafson, D.H. "Computers in clinical medicine: a critical review." _Comp. Biomed. Res._ 8, pp. 199-204 (1977).

20. Fries, J.F. "Time-oriented patient records and a computer databank." _J. Amer. Med. Assoc._ 222, pp. 1536-1542 (1972).

21. Fries, J.F. "A data bank for the clinician?" (editorial). _N. Eng. J. Med._ 294, pp. 1400-1402 (1976).

22. Garland, L.H. "Studies on the accuracy of diagnostic procedures." _Amer. J. Roentgen._ 82, pp. 25-38 (1959).

23. Gill, P.W., Leaper, D.J., Guillou, P.J., et al. "Observer variation in clinical diagnosis - a computer-aided assessment of its magnitude and importance." _Meth. Inform. Med._. 12, pp. 108-113 (1973).

24. Ginsberg, A.S. _Decision Analysis in Clinical Patient Management With an Application to the Pleural Effusion Syndrome._ The Rand Corporation, R-751-RC/NLM, July 1971.

25. Ginsberg, A.S. "The diagnostic process viewed as a decision problem." In _Computer Diagnosis and Diagnostic Methods_, (J.A. Jacquez, ed.), Springfield, Ill.: Charles C. Thomas, 1972.

26. Gleser, M.A. and Collen, M.F. "Towards automated medical decisions." _Comp. Biomed. Res._ 5, pp. 180-189 (1972).

27. Goldwyn, R.M., Friedman, H.P., Siegel, J.H. "Iteration and interaction in computer data bank analysis: as case study in the physiologic classification and assessment of the critically ill." _Comp. Biomed. Res._ 6(1973).

28. Gorry, G.A. and Barnett, G.O. "Experience with a model of sequential diagnosis." _Comp. Biomed. Res._ 1, pp. 490-507 (1968).

29. Gorry, G.A., Kassirer, J.P., Essig, A., and Schwartz, W.B. "Decision analysis as the basis for computer-aided management of acute renal failure." _Amer. J. Med_ 55, pp. 473-484 (1973).

30. Gorry, G.A. "Computer-assisted clinical decision making." _Meth. Inform. Med._ 12, pp. 45-51 (1973).

31. Gorry, G.A., Silverman, H., and Pauker, S.G. "Capturing clinical expertise: a computer program that considers clinical responses to digitalis." _Amer. J. Med_ 64, pp. 452-460 (1978).

32. Greenes, R.A., Barnett, G.O., Klein, S.W., et al. "Recording, retrieval,

and review of medical data by physician-computer interaction." N. Eng. J. Med. 282, pp. 307-315 (1970).

33. Greenfield, S., Komaroff, A.L., and Anderson, H. "A headache protocol for nurses: effectiveness and efficiency." Arch. Intern. Med. 136, pp. 1111-1116 (1976).

34. Grimm, R.H., Shimoni, K., Harlan, W.R., and Estes, E.H. "Evaluation of patient-care protocol use by various providers." N. Eng. J. Med. 292, pp. 507-511 (1975).

35. Groner, G.F., Clark, R.L., Berman, R.A., and De Land, E.C. "BIOMOD - an interactive computer graphics system for modeling." Proc. Fall Joint Computer Conference, pp. 369-378, 1971.

36. Hess, E.V. "A uniform database for rheumatic diseases." Arthritis and Rheumatism 19, pp. 645-648 (1976).

37. Hewitt, C. Description and Theoretical Analysis (Using Schemata) of PLANNER: A Language for Proving Theorems and Manipulating Models in a Robot. Ph.D. Dissertation, Department of Mathematics, Massachusetts Institute of Technology, Cambridge, Mass., 1972.

38. Horrocks, J.C., McCann, A.P., Staniland, J.R., et al. "Computer-aided diagnosis: description of an adaptable system, and operational experience with 2,034 cases." Brit. Med. J. 2, pp. 5-9 (1972).

39. Horrocks, J.C., and deDombal, F.T. "Computer-aided diagnosis of dyspepsia." Amer. J. Diges. Dis. 20,397-406 (1975).

40. Howard, R. A. (ed.). "Special Issue on Decision Analysis." IEEE Transactions on Systems, Science and Cybernetics, vol SSC-4(3), Sept., 1968.

41. Inglefinger, F.J. "Decision in medicine" (editorial). N. Eng. J. Med. 293, pp. 254-255 (1975).

42. Jacquez, J.A. Computer Diagnosis and Diagnostic Methods, Springfield, Ill.: Charles C. Thomas, 1972.

43. Jelliffe, R.W., Buell, J., Kalaba, R., et al. "A computer program for digitalis dosage regimens." Math. Biosci. 9, pp. 179-193 (1970).

44. Jelliffe, R.W., Buell, J., and Kalaba, R. "Reduction of digitalis toxicity by computer-assisted glycoside dosage regimens." Anns. Int. Med. 77, pp. 891-906 (1972).

45. Johnson, D.C. and Barnett, G.O. "MEDINFO - a medical information system." Comp. Prog. in Biomed. 7, pp. 191-201 (1977).

46. Kanal, L.N. "Patterns in Pattern Recognition: 1968-1974," IEEE Trans. on Information Theory, vol. IT-20, no. 6 (1974).

47. Karpinski, R.H.S. and Bleich, H.L. "MISAR: a miniature information storage and retrieval system." Comp. Biomed. Res. 4, pp. 655-660 (1971).

48. Kassirer, J.P. and Gorry, G.A. "Clinical problem solving: a behavioral analysis." Anns. Int. Med. 89, pp. 245-255 (1978).

49. Kleinmuntz, B. and McLean, R.S. "Diagnostic interviewing by digitalcomputer." Behav. Sci. 13, pp. 75-80 (1968).

50. Knapp, R.G., Levi, S., Lurie, D., and Westphal, M. " A computer-generated diagnostic decision guide: a comparison of statistical diagnosis and clinical diagnosis." Comput. Biol. Med. 7, pp. 223-230 (1977).

51. Komoroff, A.L., Black, W.L., Flatley, M., et al. "Protocols for physician assistants: management of diabetes and hypertension." N. Eng. J. Med. 290,307-312 (1974).

52. Korein, J., Lyman, M., and Tick, J.L. " The computerized medical record," Bulletin New York Academy of Medicine, Vol.47, pp. 824-826 (1971).

53. Koss, N. and Feinstein, A.R. "Computer-aided prognosis: II. development of a prognostic algorithm." Arch. Intern. Med. 127, pp. 448-459 (1971).

54. Leaper, D.J., Horrocks, J.C., Staniland, J.P., and deDombal, F.T. "Computer-assisted diagnosis of abdominal pain using 'estimates' provided by clinicians." Brit. Med. J. 4, pp. 350-354 (1972).

55. Ledley, R.S. and Lusted, L.B. "Reasoning foundations of medical diagnosis." Science 130,9-21 (1959).

56. Levi, S., Frant, J.R., Westphal, M.C., and Lurie, D. "Development of a decision guide - optimal discriminations for meningitis determined by statistical analysis." Meth. Inform. Med. 15 (2), 87-90 (1976).

57. Lipkin, M. and Hardy, J.D. "Mechanical correlation of data in differential diagnosis of hematologic diseases." J. Amer. Med. Assoc. 166, pp. 113-125 (1958).

58. Lusted, L.B. Introduction To Medical Decision Making. Springfield, Ill.: Charles C. Thomas, 1968.

59. Mabry, J.C., Thompson, H.K., Hopwood, M.D., and Baker, W.R. "A prototype data management and analysis system (CLINFO): system description and user experience." In MEDINFO 77, Amsterdam: North-Holland Publishing Co., 1977, pp. 71-75.

60. McDonald, C., Bhargava, B., and Jeris, D. "A clinical information system (CIS) for ambulatory care," Proc. of the 1975 NCC, AFIPS Press, vol. 44 (1975) pp. 749-756

61. McNeil, B.J., Keeler, E., and Adelstein, S.J. "Primer on certain elements of medical decision making." N. Eng. J. Med. 293, pp. 211-215 (1975).

62. McNeil, B.J. and Adelstein, S.J. "Determining the value of diagnostic and screening tests." J. Nucl. Med. 17, pp. 439-448 (1977).

63. Menn, S.J., Barnett, G.O., Schmechel, D., et al. "A computer program to assist in the care of acute respiratory failure." J. Amer. Med. Assoc. 223, pp. 308-312 (1973).

64. Mesel, E., Wirtschafter, D.D., Carpenter, J.T., et al. Clinical algorithms for cancer chemotherapy - systems for community-based consultant-extenders and oncology centers. Meth. Inform. Med. 15, pp. 168-173 (1976).

65. Nordyke, R.A., Kulikowski, C.A., and Kulikowski, C.W. "A comparison of methods for the automated diagnosis of thyroid dysfunction." Comp. Biomed. Res. 4, pp. 374-389 (1971).

66. Norusis, M.J. and Jacquez, J.A. "Diagnosis. I. Symptom nonindependence in mathematical models for diagnosis." Comp. Biomed. Res. 8, pp. 156-172 (1975).

67. Patrick, E.A. "Pattern Recognition in Medicine," Systems, Man and Cybernetics Review, 6, p. 4 (1977).

68. Pauker, S.G. and Kassirer, J.P. "Therapeutic decision making: a cost-benefit analysis." N. Eng. J. Med. 293, pp. 229-234 (1975).

69. Pauker, S.G., Gorry, G.A., Kassirer, J.P., and Schwartz, W.B. "Towards the simulation of clinical cognition: taking a present illness by computer." Amer. J. Med. 60:981-996 (1976).

70. Pauker, S.G. "Coronary artery surgery: the use of decision analysis." Anns. Int. Med. 85, pp. 8-18 (1976).

71. Pauker, S.P. and Pauker, S.G. "Prenatal diagnosis: a directive approach to genetic counseling using decision analysis." Yale J. Biol. Med. 50,275-289 (1977).

72. Peck, C.C., Sheiner, L.B., Martin, C.M., et al. "Computer-assisted digoxin therapy." N. Eng. J. Med. 289, pp. 441-446 (1973).

73. Pipberger, H.V. "Clinical application of a second generation electrocardiography computer program." Amer. J. Electrocardiology 35, pp. 597-608 (1975).

74. Pliskin, J.S. and Beck, C.H. "Decision analysis in individual clinical decision making: a real-world application in treatment of renal disease." Meth. Inform. Med. 15, pp. 43-46 (1976).

75. Pople, H.E., Myers, J.D. and Miller, R.A. "DIALOG: A model of diagnostic logic for internal medicine." Proc. 4th Int. Joint. Conf. on Artif. Intell., MIT, Cambridge, Mass., 1975.

76. Pople, H. "The formation of composite hypotheses in diagnostic problem solving: an exercise in synthetic reasoning." Proc. of 5th Intl Joint Conf on Artif. Intelligence, Cambridge, Mass, 1977, pp. 1030-1037.

77. Prutting, J. "Lack of correlation between antemortem and postmortem diagnosis." N.Y. J. Med. 67, pp. 2081-2084 (1967).

78. Raiffa, H. Decision Analysis: Introductory Lectures on Choices Under Uncertainty. Reading, Mass.: Addison Wesley, 1968.

79. Richards, B. and Goh, A.E.S. "Computer assistance in the treatment of patients with acid-base and electrolyte disturbances." MEDINFO 77, Amsterdam: North-Holland Publishing Company, 1977, pp. 407-410.

80. Rodnick, J., and Wiederhold, G. , "Review of automated ambulatory medical record systems: charting services that are of essential benefit to the physician," MEDINFO 77, Amsterdam: North-Holland Publishing Co., 1977, pp. 957-961.

184

81. Rosati, R.A., Wallace, A.G., and Stead, E.A. "The way of the future." Arch. Intern. Med. 131, pp. 285-287 (1973).

82. Rosati, R.D., McNeer, J.F., Starmer, C.F., et al. "A new information system for medical practice." Arch. Intern. Med. 135, pp. 1017-1024 (1975).

83. Rosenblatt, M.B., Teng, P.K., and Kerpe, S. "Diagnostic accuracy in cancer as determined by post-mortem examination." Prog. Clin. Cancer 5, pp. 71-80 (1973).

84. Rubin, A.D. and Risley, J.F. "The PROPHET system: an experiment in providing a computer resource to scientists." MEDINFO 77, Amsterdam: North-Holland Publishing Co., 1977, pp. 77-81.

85. Safran, C., Tsichlis, P.N., Bluming, A.Z., and Desforges, J.F. "Diagnostic planning using computer-assisted decision making for patients with Hodgkins' disease." Cancer 39, pp. 2426-2434 (1977).

86. Schoolman, H. and Bernstein, L. "Computer use in diagnosis, prognosis, and therapy." Science 200, pp. 926-931 (1978).

87. Schwartz, W.B., Gorry, G.A., Kassirer, J.P., and Essig, A. "Decision analysis and clinical judgment." Amer. J. Med 55, pp. 459-472 (1973).

88. Scott, A.C., Clancey, W., Davis, R., and Shortliffe, E.H. "Explanation capabilities of knowledge-based production systems." Amer. J. Computational Linguistics, Microfiche 62, 1977.

89. Sheiner, L.B., Halkin, H., Peck, C., et al. "Improved computer-assisted digoxin therapy." Anns. Int. Med. 82, pp. 619-627 (1975).

90. Sherman, H., Reiffen, B., and Komoroff, A.L. "Ambulatory care systems." In Problem-Directed and Medical Information Systems (M.F. Driggs, ed.), New York: Intercontinental Medical Book Corporation, 1973, pp. 143-171.

91. Shimura, M. "Learning procedures in pattern classifiers - introduction and survey." Proc. Intl. Joint Conf. on Pattern Recognition, Kyoto, 1978, pp. 125-138.

92. Shortliffe, E.H., Axline, S.G., Buchanan, B.G., and Cohen, S.N. "Design considerations for a program to provide consultations in clinical therapeutics." Proc. 13th San Diego Biomedical Symposium, 311-319, San Diego, Calif., February 1974.

93. Shortliffe, E.H. and Davis, R. "Some considerations for the implementation of knowledge-based expert systems." SIGART Newsletter, No. 55, 9-12, December 1975.

94. Shortliffe, E.H., and Buchanan, B.G. "A model of inexact reasoning in medicine." Math. Biosci. 23, pp. 351-379 (1975).

95. Shortliffe, E.H. Computer-Based Medical Consultations: MYCIN, New York: Elsevier/North Holland, 1976.

96. Slamecka, V., Camp, H.N., Badre, A.N., and Fall, V.D. "MARIS: a knowledge system for internal medicine." Inform. Process & Man. 13, pp. 273-276 (1977).

97. Sox, H.C., Sox, C.H., and Tompkins, R.K. "The training of physicians' assistants: the use of a clinical algorithm system." N. Eng. J. Med. 288, pp. 818-824 (1973).

98. Sridharan, N.S. Guest editorial. Artificial Intelligence 11, pp. 1-4 (1978).

99. Startsman, T.S., and Robinson, R.E. "The attitudes of medical and paramedical personnel towards computers." Comp. Biomed. Res. 5, pp. 218-227 (1972).

100. Stead, W.W., Brame, R.G., Hammond, W.E., et al. "A computerized obstetric medical record." Obstet. & Gyn. 49, pp. 502-509 (1977).

101. Szolovits, P. and Pauker, S.G. "Categorical and probabilistic reasoning in medical diagnosis." Artificial Intelligence 11, pp. 115-144 (1978).

102. Taylor, T.R. "Clinical decision analysis." Meth. Inform. Med. 15, pp. 216-224 (1976).

103. Vickery, D.M. "Computer support of paramedical personnel: the question of quality control." MEDINFO 74, Amsterdam: North-Holland Publishing Company, 1974, pp. 281-287.

104. Wagner, G., Tautu, P., and Wolber, U. "Problems of medical diagnosis: a bibliography." Meth. Info. Med. 17, pp. 55-74 (1978).

105. Walsh, B.T., Bookhein, W.W., Johnson, R.C., et al. "Recognition of streptococcal pharyngitis in adults." Arch. Int. Med. 135, pp. 1493-1497 (1975).

106. Wardle, A. and Wardle, L. "Computer-aided diagnosis: a review of research." Meth. Info. Med. 17, pp. 15-28 (1978).

107. Warner, H.R., Toronto, A.F., and Veasy, L.G. "Experience with Bayes' Theorem for computer diagnosis of congenital heart disease." Anns. N.Y. Acad. Sci. 115, pp. 558-567 (1964).

108. Warner, H.R. "Experiences with computer-based patient monitoring." Anes. & Analgesia Current Researchers 47, pp. 453-461 (1968).

109. Warner, H.R., Olmsted, C.M., and Rutherford, B.D. "HELP - a program for medical decision-making." Comp. Biomed. Res. 5, pp. 65-74 (1972).

110. Warner, H.R., Rutherford, B.D., and Houtchens, B. "A sequential approach to history taking and diagnosis." Comp. Biomed. Res. 5, pp. 256-262 (1972).

111. Warner, H.R., Morgan, J.D., Pryor, T.A., et al. "HELP - a self-improving system for medical decision making." MEDINFO 74, Amsterdam: North-Holland Publishing Company, 1974.

112. Warner, H.R. Knowledge sectors for logical processing of patient data in the HELP system." Proc. of 2nd. Ann. Symp. on Computer Applications in Medical Care, IEEE, Wash. D.C.,(1978), pp. 401-404.

113. Watson, R.J. "Medical staff response to a medical information system with

direct physician-computer interface." MEDINFO 74, p. 299-302, Amsterdam: North-Holland Publishing Company, 1974.

114. Wechsler, H. "A fuzzy approach to medical diagnosis." Int. J. Biomed. Comp. 7, pp. 191-203 (1976).

115. Weed, L.L. "Medical records that guide and teach." N. Eng. J. Med. 278, pp. 593-599,652-657 (1968).

116. Weed, L.L. "Problem-oriented medical records." In Problem-Directed and Medical Information Systems (M.F. Driggs, ed.), New York: Intercontinental Medical Book Corporation, 1973.

117. Weiss, S.M., Kulikowski, C.A., Amarel, S. and Safir, A. "A model-based method for computer-aided medical decision-making." Artificial Intelligence 11, pp. 145-172 (1978).

118. Weyl, S., Fries, J., Wiederhold, G., and Germano, F. "A modular self-describing clinical databank system." Comp. Biomed. Res. 8, pp. 279-293 (1975).

119. Wiederhold, G., Fries, J.F., and Weyl, S. "Structured organization of clinical databases," Proc. of the 1975 NCC, AFIPS Press vol. 44 (1975) pp. 479-485

120. Winston, P.H. Artificial Intelligence, Reading, Mass.: Addison-Wesley, 1977.

121. Wortman, P.M. "Medical diagnosis: an information processing approach." Comput. Biomed. Res. 5, pp. 315-328 (1972).

122. Yu, V.L., Fagan, L.M., Wraith, S.M., et al. "Computer-based consultation in antimicrobial selection - a comparative evaluation by experts." Stanford University School of Medicine. Submitted for publication, November 1978.

123. Yu, V.L., Buchanan, B.G., Shortliffe, E.H., et al. "An evaluation of the performance of a computer-based consultant." To appear in Comput. Prog. Biomed., 1979.

124. Zadeh, L.A. "Fuzzy sets." Information and Control 8, pp. 338-353 (1965).

125. Zoltie, N., Horrocks, J.C., and deDombal, F.T. "Computer-assisted diagnosis of dyspepsia - report on transferability of a system, with emphasis on early diagnosis of gastric cancer." Meth. Inform. Med. 16, pp. 89-92 (1977).

THE ART OF ARTIFICIAL INTELLIGENCE:

I. Themes and case studies of knowledge engineering

Edward A. Feigenbaum

Department of Computer Science,
Stanford University,
Stanford, California, 94305.

Abstract

The knowledge engineer practices the art of bringing the principles and tools of AI research to bear on difficult applications problems requiring experts' knowledge for their solution. The technical issues of acquiring this knowledge, representing it, and using it appropriately to construct and explain lines-of-reasoning, are important problems in the design of knowledge-based systems. Various systems that have achieved expert level performance in scientific and medical inference illuminate the art of knowledge engineering and its parent science, Artificial Intelligence.

INTRODUCTION: AN EXAMPLE

This is the first of a pair of papers that will examine emerging themes of knowledge engineering, illustrate them with case studies drawn from the work of the Stanford Heuristic Programming Project, and discuss general issues of knowledge engineering art and practice.

Let me begin with an example new to our workbench: a system called PUFF, the early fruit of a collaboration between our project and a group at the Pacific Medical Center (PMC) in San Francisco.

A physician refers a patient to PMC's pulmonary function testing lab for diagnosis of possible pulmonary function disorder. For one of the tests, the patient inhales and exhales a few times in a tube connected to an instrument/computer combination. The instrument acquires data on flow rates and volumes, the so-called flow-volume loop of the patient's lungs and airways. The computer measures certain parameters of the curve and presents them to the diagnostician (physician or PUFF) for interpretation. The diagnosis is made along these lines: normal or diseased; restricted lung disease or obstructive airways disease or a combination of both; the severity; the likely disease type(s) (e.g. emphysema, bronchitis, etc.); and other factors important for diagnosis.

PUFF is given not only the measured data but also certain items of information from the patient record, e.g. sex, age, number of pack-years of cigarette smoking. The task of the PUFF system is to infer a diagnosis and print it out in English in the normal medical summary form of the interpretation expected by the referring physician.

Everything PUFF knows about pulmonary function diagnosis is contained in (currently) 55 rules of the IF...THEN... form. No textbook of medicine currently records these rules. They constitute the partly-public, partly-private knowledge of an expert pulmonary physiologist at PMC, and were extracted and polished by project engineers working intensively with the expert over a period of time. Here is an example of a PUFF rule (the unexplained acronyms refer to various data measurements):

RULE 31

IF:
1) The severity of obstructive airways disease of the patient is greater than or equal to mild, and
2) The degree of diffusion defect of the patient is greater than or equal to mild, and
3) The tlc(body box)observed/predicted of the patient is greater than or equal to 110 and
4) The observed-predicted difference in rv/tlc of the patient is greater than or equal to 10

THEN:
1) There is strongly suggestive evidence (.9) that the subtype of obstructive airways disease is emphysema, and
2) It is definite (1.0) that "OAD, Diffusion Defect, elevated TLC, and elevated RV together indicate emphysema." is one of the findings.

One hundred cases, carefully chosen to span the variety of disease states with sufficient exemplary information for each, were used to extract the 55 rules. As the knowledge emerged, it was represented in rule form, added to the system and tested by running additional cases. The expert was sometimes surprised, sometimes frustrated, by the occasional gaps and inconsistencies in the knowledge, and the incorrect diagnoses that were logical consequences of the existing rule set. The interplay between knowledge engineer and expert gradually expanded the set of rules to remove most of these problems.

As cumulation of techniques in the art demands and allows, a new tool was not invented when an old one would do. The knowledge engineers pulled out of their toolkit a version of the MYCIN system (to be discussed later), with the rules about infectious diseases removed, and used it as the inference engine for the PUFF diagnoses. Thus PUFF, like MYCIN, is a relatively simple backward-chaining inference system. It seeks a valid line-of-reasoning based on its rules and rooted in the instrument and patient data. With a little more work at fitting some existing tools together, PUFF will be able to explain this line-of-reasoning, just as MYCIN does.

As it is, PUFF only prints out the final interpretation, of which the following is an example:

PATIENT DATA:
The degree of dyspnea: MODERATELY-SEVERE
The severity of coughing: MILD
Sputum production MODERATELY-SEVERE
The number of pack-years of smoking: 48
referral diagnosis: BRONCHITIS
IVC/IVC-predicted: 80
RV/RV-predicted: 191
FVC/FVC-predicted: 87
TLC(body box)observed/predicted: 127
Predicted FEVl/FVC: 83
TLC(DLCO)observed/predicted: 83
FEVl/FVC ratio: 50
RV/TLC Observed-Predicted: 21
MMF/MMF-predicted: 19
the DLCO/DLCO-predicted: 48
The slope (F50_obs-F25_obs)/FVC_obs: 19

DEGREE OF OBSTRUCTIVE AIRWAYS DISEASE:

OAD degree by SLOPE: (MODERATELY-SEVERE 700)
OAD degree by MMF: (SEVERE 900)
OAD degree by FEVl: (MODERATELY-SEVERE 700)

FINAL OAD DEGREE: (MODERATELY-SEVERE 910) (SEVERE 900)
No conflict. Final degree:
(MODERATELY-SEVERE 910)

INTERPRETATION:

Obstruction is indicated by curvature of the flow-volume loop.
Forced Vital Capacity is normal and peak flow rates are reduced, suggesting airway obstruction.
Flow rate from 25-75 of expired volume is reduced, indicating severe airway obstruction.
OAD, Diffusion Defect, elevated TLC, and elevated RV together indicate emphysema.
OAD, Diffusion Defect, and elevated RV indicate emphysema.
Change in expired flow rates following bronchodilation shows that there is reversibility of airway obstruction.
The presence of a productive cough is an indication that the OAD is of the bronchitic type.
Elevated lung volumes indicate overinflation.
Air trapping is indicated by the elevated difference between observed and predicted RV/TLC ratios.
Improvement in airway resistance indicates some reversibility of airway
Airway obstruction is consistent with the patient's smoking history.
The airway obstruction accounts for the patient's dyspnea.
Although bronchodilators were not useful in this one case, prolonged use may prove to be beneficial to the patient.
The reduced diffusion capacity indicates airway obstruction of the mixed bronchitic and emphysematous types.
Low diffusing capacity indicates loss of alveolar capillary surface.
Obstructive Airways Disease of mixed types

---

150 cases not studied during the knowledge acquisition process were used for a test and validation of the rule set. PUFF inferred a diagnosis for each. PUFF-produced and expert-produced interpretations were coded for statistical analysis to discover the degree of agreement. Over various types of disease states, and for two conditions of match between human and computer diagnoses ("same degree of severity" and "within one degree of severity"), agreement ranged between approximately 90% and 100%.

The PUFF story is just beginning and will be told perhaps at the next IJCAI. The surprising punchline to my synopsis is that the current state of the PUFF system as described above was achieved in less than 50 hours of interaction with the expert and less than 10 man-weeks of effort by the knowledge engineers. We have learned much in the

past decade of the art of engineering knowledge-based intelligent agents!

In the remainder of this essay, I would like to discuss the route that one research group, the Stanford Heuristic Programming Project, has taken, illustrating progress with case studies, and discussing themes of the work.

2 ARTIFICIAL INTELLIGENCE & KNOWLEDGE ENGINEERING

The dichotomy that was used to classify the collected papers in the volume Computers and Thought still characterizes well the motivations and research efforts of the AI community. First, there are some who work toward the construction of intelligent artifacts, or seek to uncover principles, methods, and techniques useful in such construction. Second, there are those who view artificial intelligence as (to use Newell's phrase) "theoretical psychology," seeking explicit and valid information processing models of human thought.

For purposes of this essay, I wish to focus on the motivations of the first group, these days by far the larger of the two. I label these motivations "the intelligent agent viewpoint" and here is my understanding of that viewpoint:

"The potential uses of computers by people to accomplish tasks can be 'one-dimensionalized' into a spectrum representing the nature of instruction that must be given the computer to do its job. Call it the WHAT-TO-HOW spectrum. At one extreme of the spectrum, the user supplies his intelligence to instruct the machine with precision exactly HOW to do his job, step-by-step. Progress in Computer Science can be seen as steps away from the extreme 'HOW' point on the spectrum: the familiar panoply of assembly languages, subroutine libraries, compilers, extensible languages, etc. At the other extreme of the spectrum is the user with his real problem (WHAT he wishes the computer, as his instrument, to do for him). He aspires to communicate WHAT he wants done in a language that is comfortable to him (perhaps English); via communication modes that are convenient for him (including perhaps, speech or pictures); with some generality, some vagueness, imprecision, even error; without having to lay out in detail all necessary subgoals for adequate performance - with reasonable assurance that he is addressing an intelligent agent that is using knowledge of his world to understand his intent, to fill in his vagueness, to make specific his abstractions, to correct his errors, to discover appropriate subgoals, and

ultimately to translate WHAT he really wants done into processing steps that define HOW it shall be done by a real computer. The research activity aimed at creating computer programs that act as "intelligent agents" near the WHAT end of the WHAT-TO-HOW spectrum can be viewed as the long-range goal of AI research." (Feigenbaum, 1974)

Our young science is still more art than science. Art: "the principles or methods governing any craft or branch of learning." Art: "skilled workmanship, execution, or agency." These the dictionary teaches us. Knuth tells us that the endeavor of computer programming is an art, in just these ways. The art of constructing intelligent agents is both part of and an extension of the programming art. It is the art of building complex computer programs that represent and reason with knowledge of the world. Our art therefore lives in symbiosis with the other worldly arts, whose practitioners -- experts of their art — hold the knowledge we need to construct intelligent agents. In most "crafts or branches of learning" what we call "expertise" is the essence of the art. And for the domains of knowledge that we touch with our art, it is the "rules of expertise" or the rules of "good judgment" of the expert practitioners of that domain that we seek to transfer to our programs.

2.1 Lessons of the Past

Two insights from previous work are pertinent to this essay.

The first concerns the quest for generality and power of the inference engine used in the performance of intelligent acts (what Minsky and Papert [see Goldstein and Papert, 1977] have labeled "the power strategy"). We must hypothesize from our experience to date that the problem solving power exhibited in an intelligent agent's performance is primarily a consequence of the specialist's knowledge employed by the agent, and only very secondarily related to the generality and power of the inference method employed. Our agents must be knowledge-rich, even if they are methods-poor. In 1970, reporting the first major summary-of-results of the DENDRAL program (to be discussed later), we addressed this issue as follows:

"...general problem-solvers are too weak to be used as the basis for building high-performance systems. The behavior of the best general problem-solvers we know, human problem-solvers, is observed to be weak and shallow, except in the areas in which the human problem-solver is a specialist. And it is observed that the transfer of expertise between specialty

areas is slight. A chess master is unlikely to be an expert algebraist or an expert mass spectrum analyst, etc. In this view, the expert is the specialist, with a specialist's knowledge of his area and a specialist's methods and heuristics." (Feigenbaum, Buchanan and Lederberg, 1971, p. 187)

Subsequent evidence from our laboratory and all others has only confirmed this belief.

AI researchers have dramatically shifted their view on generality and power in the past decade. In 1967, the canonical question about the DENDRAL program was: "It sounds like good chemistry, but what does it have to do with AI?" In 1977, Goldstein and Papert write of a paradigm shift in AI:

"Today there has been a shift in paradigm. The fundamental problem of understanding intelligence is not the identification of a few powerful techniques, but rather the question of how to represent large amounts of knowledge in a fashion that permits their effective use and interaction." (Goldstein and Papert, 1977)

The second insight from past work concerns the nature of the knowledge that an expert brings to the performance of a task. Experience has shown us that this knowledge is largely heuristic knowledge, experiential, uncertain — mostly "good guesses" and "good practice," in lieu of facts and rigor. Experience has also taught us that much of this knowledge is private to the expert, not because he is unwilling to share publicly how he performs, but because he is unable. He knows more than he is aware of knowing. [Why else is the Ph.D. or the Internship a guild-like apprenticeship to a presumed "master of the craft?" What the masters really know is not written in the textbooks of the masters.] But we have learned also that this private knowledge can be uncovered by the careful, painstaking analysis of a second party, or sometimes by the expert himself, operating in the context of a large number of highly specific performance problems. Finally, we have learned that expertise is multi-faceted, that the expert brings to bear many and varied sources of knowledge in performance. The approach to capturing his expertise must proceed on many fronts simultaneously.

2.2 The Knowledge Engineer

The knowledge engineer is that second party just discussed. [An historical note about the term. In the mid-60s, John McCarthy, for reasons obvious from his work, had been describing Artificial Intelligence as "Applied Epistemology." When I first described the DENDRAL program to Donald Michie in 1968, he remarked that it was "epistemological engineering," a clever but ponderous and unpronounceable turn-of-phrase that I simplified into "knowledge engineering."] She (in deference to my favorite knowledge engineer) works intensively with an expert to acquire domain-specific knowledge and organize it for use by a program. Simultaneously she is matching the tools of the AI workbench to the task at hand — program organizations, methods of symbolic inference, techniques for the structuring of symbolic information, and the like. If the tool fits, or nearly fits, she uses it. If not, necessity mothers AI invention, and a new tool gets created. She builds the early versions of the intelligent agent, guided always by her intent that the program eventually achieve expert levels of performance in the task. She refines or reconceptualizes the system as the increasing amount of acquired knowledge causes the AI tool to "break" or slow down intolerably. She also refines the human interface to the intelligent agent with several aims: to make the system appear "comfortable" to the human user in his linguistic transactions with it; to make the system's inference processes understandable to the user; and to make the assistance controllable by the user when, in the context of a real problem, he has an insight that previously was not elicited and therefore not incorporated.

In the next section, I wish to explore (in summary form) some case studies of the knowledge engineer's art.

3 CASES FROM THE KNOWLEDGE ENGINEER'S WORKSHOP

I will draw material for this section from the work of my group at Stanford. Much exciting work in knowledge engineering is going on elsewhere. Since my intent is not to survey literature but to illustrate themes, at the risk of appearing parochial I have used as case studies the work I know best.

My collaborators (Professors Lederberg and Buchanan) and I began a series of projects, initially the development of the DENDRAL program, in 1965. We had dual motives: first, to study scientific problem solving and discovery, particularly the processes scientists do use or should use in inferring hypotheses and theories from empirical evidence; and second, to conduct this study in such a way that our experimental programs would one day be of use to working scientists, providing intelligent assistance on important and difficult problems. By 1970, we and

our co-workers had gained enough experience that
we felt comfortable in laying out a program of
research encompassing work on theory formation,
knowledge utilization, knowledge acquisition,
explanation, and knowledge engineering techniques.
Although there were some surprises along the way
(like the AM program), the general lines of the
research are proceeding as envisioned.

THEMES

As a road map to these case studies, it is
useful to keep in mind certain major themes:

Generation-and-test: Omnipresent in our
experiments is the "classical" generation-and-
test framework that has been the hallmark of AI
programs for two decades. This is not a
consequence of a doctrinaire attitude on our part
about heuristic search, but rather of the
usefulness and sufficiency of the concept.

Situation => Action Rules: We have chosen to
represent the knowledge of experts in this form.
Making no doctrinaire claims for the universal
applicability of this representation, we
nonetheless point to the demonstrated utility of
the rule-based representation. From this
representation flow rather directly many of the
characteristics of our programs: for example,
ease of modification of the knowledge, ease of
explanation. The essence of our approach is that
a rule must capture a "chunk" of domain knowledge
that is meaningful, in and of itself, to the
domain specialist. Thus our rules bear only a
historical relationship to the production rules
used by Newell and Simon (1972) which we view as
"machine-language programming" of a
recognize => act machine.

The Domain-Specific Knowledge: It plays a critical
role in organizing and constraining search. The
theme is that in the knowledge is the power. The
interesting action arises from the knowledge
base, not the inference engine. We use knowledge
in rule form (discussed above), in the form of
inferentially-rich models based on theory, and in
the form of tableaus of symbolic data and
relationships (i.e. frame-like structures).
System processes are made to conform to natural
and convenient representations of the domain-
specific knowledge.

Flexibility to modify the knowledge base: If the
so-called "grain size" of the knowledge
representation is chosen properly (i.e. small
enough to be comprehensible but large enough to
be meaningful to the domain specialist), then the
rule-based approach allows great flexibility for
adding, removing, or changing knowledge in the
system.

Line-of-reasoning: A central organizing principle
in the design of knowledge-based intelligent
agents is the maintenance of a line-of-reasoning
that is comprehensible to the domain specialist.

This principle is, of course, not a logical
necessity, but seems to us to be an engineering
principle of major importance.

Multiple Sources of Knowledge: The formation and
maintenance (support) of the line-of-reasoning
usually require the integration of many disparate
sources of knowledge. The representational and
inferential problems in achieving a smooth and
effective integration are formidable engineering
problems.

Explanation: The ability to explain the line-of-
reasoning in a language convenient to the user is
necessary for application and for system
development (e.g. for debugging and for extending
the knowledge base). Once again, this is an
engineering principle, but very important. What
constitutes "an explanation" is not a simple
concept, and considerable thought needs to be
given, in each case, to the structuring of
explanations.

CASE STUDIES

In this section I will try to illustrate
these themes with various case studies.

3.1 DENDRAL: Inferring Chemical Structures

3.1.1 Historical Note

Begun in 1965, this collaborative project
with the Stanford Mass Spectrometry Laboratory has
become one of the longest-lived continuous efforts
in the history of AI (a fact that in no small way
has contributed to its success). The basic
framework of generation-and-test and rule-based
representation has proved rugged and extendable.
For us the DENDRAL system has been a fountain of
ideas, many of which have found their way, highly
metamorphosed, into our other projects. For
example, our long-standing commitment to rule-
based representations arose out of our
(successful) attempt to head off the imminent
ossification of DENDRAL caused by the rapid
accumulation of new knowledge in the system around
1967.

3.1.2 Task

To enumerate plausible structures (atom-bond
graphs) for organic molecules, given two kinds of
information: analytic instrument data from a mass
spectrometer and a nuclear magnetic resonance
spectrometer; and user-supplied constraints on the
answers, derived from any other source of
knowledge (instrumental or contextual) available
to the user.

3.1.3 Representations

Chemical structures are represented as node-link graphs of atoms (nodes) and bonds (links). Constraints on search are represented as subgraphs (atomic configurations) to be denied or preferred. The empirical theory of mass spectrometry is represented by a set of rules of the general form:

```
Situation: Particular atomic
           configuration
           (subgraph)

                 |
                 | Probability, P,
                 | of occurring
                 |
                 V

Action:    Fragmentation of the
           particular configuration
           (breaking links)
```

Rules of this form are natural and expressive to mass spectrometrists.

3.1.4 Sketch of Method

DENDRAL's inference procedure is a heuristic search that takes place in three stages, without feedback: plan-generate-test.

"Generate" (a program called CONGEN) is a generation process for plausible structures. Its foundation is a combinatorial algorithm (with mathematically proven properties of completeness and non-redundant generation) that can produce all the topologically legal candidate structures. Constraints supplied by the user or by the "Plan" process prune and steer the generation to produce the plausible set (i.e. those satisfying the constraints) and not the enormous legal set.

"Test" refines the evaluation of plausibility, discarding less worthy candidates and rank-ordering the remainder for examination by the user. "Test" first produces a "predicted" set of instrument data for each plausible candidate, using the rules described. It then evaluates the worth of each candidate by comparing its predicted data with the actual input data. The evaluation is based on heuristic criteria of goodness-of-fit. Thus, "test" selects the "best" explanations of the data.

"Plan" produces direct (i.e. not chained) inference about likely substructure in the molecule from patterns in the data that are indicative of the presence of the substructure. (Patterns in the data trigger the left-hand-sides

of substructure rules). Though composed of many atoms whose interconnections are given, the substructure can be manipulated as atom-like by "generate." Aggregating many units entering into a combinatorial process into fewer higher-level units reduces the size of the combinatorial search space. "Plan" sets up the search space so as to be relevent to the input data. "Generate is the inference tactician; "Plan" is the inference strategist. There is a separate "Plan" package for each type of instrument data, but each package passes substructures (subgraphs) to "Generate." Thus, there is a uniform interface between "Plan" and "Generate." User-supplied constraints enter this interface, directly or from user-assist packages, in the form of substructures.

3.1.5 Sources of Knowledge

The various sources of knowledge used by the DENDRAL system are:

Valences (legal connections of atoms); stable and unstable configurations of atoms; rules for mass spectrometry fragmentations; rules for NMR shifts; expert's rules for planning and evaluation; user-supplied constraints (contextual).

3.1.6 Results

DENDRAL's structure elucidation abilities are, paradoxically, both very general and very narrow. In general, DENDRAL handles all molecules, cyclic and tree-like. In pure structure elucidation under constraints (without instrument data),CONGEN is unrivaled by human performance. In structure elucidation with instrument data, DENDRAL's performance rivals expert human performance only for a small number of molecular families for which the program has been given specialist's knowledge, namely the families of interest to our chemist collaborators. I will spare this computer science audience the list of names of these families. Within these areas of knowledge-intensive specialization, DENDRAL's performance is usually not only much faster but also more accurate than expert human performance.

The statement just made summarizes thousands of runs of DENDRAL on problems of interest to our experts, their colleagues, and their students. The results obtained, along with the knowledge that had to be given to DENDRAL to obtain them, are published in major journals of chemistry. To date, 25 papers have been published there, under a series title "Applications of Artificial Intelligence for Chemical Inference: <specific subject>" (see references).

The DENDRAL system is in everyday use by Stanford chemists, their collaborators at other universities and collaborating or otherwise interested chemists in industry. Users outside

Stanford access the system over commercial computer/communications network. The problems they are solving are often difficult and novel. The British government is currently supporting work at Edinburgh aimed at transferring DENDRAL to industrial user communities in the UK.

3.1.7 Discussion

Representation and extensibility. The representation chosen for the molecules, constraints, and rules of instrument data interpretation is sufficiently close to that used by chemists in thinking about structure elucidation that the knowledge base has been extended smoothly and easily, mostly by chemists themselves in recent years. Only one major reprogramming effort took place in the last 9 years -- when a new generator was created to deal with cyclic structures.

Representation and the Integration of multiple sources of knowledge. The generally difficult problem of integrating various sources of knowledge has been made easy in DENDRAL by careful engineering of the representations of objects, constraints, and rules. We insisted on a common language of compatibility of the representations with each other and with the inference processes: the language of molecular structure expressed as graphs. This leads to a straightforward procedure for adding a new source of knowledge, say, for example, the knowledge associated with a new type of instrument data. The procedure is this: write rules that describe the effect of the physical processes of the instrument on molecules using the situation => action form with molecular graphs on both sides; any special inference process using these rules must pass its results to the generator only(!) in the common graph language.

It is today widely believed in AI that the use of many diverse sources of knowledge in problem solving and data interpretation has a strong effect on quality of performance. How strong is, of course, domain-dependent, but the impact of bringing just one additional source of knowledge to bear on a problem can be startling. In one difficult (but not unusually difficult) mass spectrum analysis problem*, the program using its mass spectrometry knowledge alone would have generated an impossibly large set of plausible candidates (over 1.25 million!). Our engineering response to this was to add another source of data and knowledge, proton NMR. The addition on a simple interpretive theory of this NMR data, from which the program could infer a few additional constraints, reduced the set of plausible candidates to one, the right structure! This was not an isolated result but showed up dozens of times in subsequent analyses.

----
* the analysis of an acyclic amine with formula $C20H45N$.

DENDRAL and data. DENDRAL's robust models (topological, chemical, instrumental) permit a strategy of finding solutions by generating hypothetical "correct answers" and choosing among these with critical tests. This strategy is opposite to that of piecing together the implications of each data point to form a hypothesis. We call DENDRAL's strategy largely model-driven, and the other data-driven. The consequence of having enough knowledge to do model-driven analysis is a large reduction in the amount of data that must be examined since data is being used mostly for verification of possible answers. In a typical DENDRAL mass spectrum analysis, usually no more than about 25 data points out of a typical total of 250 points are processed. This important point about data reduction and focus-of-attention has been discussed before by Gregory (1968) and by the vision and speech research groups, but is not widely understood.

Conclusion. DENDRAL was an early herald of AI's shift to the knowledge-based paradigm. It demonstrated the point of the primacy of domain-specific knowledge in achieving expert levels of performance. Its development brought to the surface important problems of knowledge representation, acquisition, and use. It showed that, by and large, the AI tools of the first decade were sufficient to cope with the demands of a complex scientific problem-solving task, or were readily extended to handle unforseen difficulties. It demonstrated that AI's conceptual and programming tools were capable of producing programs of applications interest, albeit in narrow specialties. Such a demonstration of competence and sufficiency was important for the credibility of the AI field at a critical juncture in its history.

3.2 META-DENDRAL: inferring rules of mass spectrometry

3.2.1 Historical note

The META-DENDRAL program is a case study in automatic acquisition of domain knowledge. It arose out of our DENDRAL work for two reasons: first, a decision that with DENDRAL we had a sufficiently firm foundation on which to pursue our long-standing interest in processes of scientific theory formation; second, by a recognition that the acquisition of domain knowledge was the bottleneck problem in the building of applications-oriented intelligent agents.

3.2.2 Task

META-DENDRAL's job is to infer rules of fragmentation of molecules in a mass spectrometer for possible later use by the DENDRAL performance

program. The inference is to be made from actual spectra recorded from known molecular structures. The output of the system is the set of fragmentation rules discovered, summary of the evidence supporting each rule, and a summary of contra-indicating evidence. User-supplied constraints can also be input to force the form of rules along desired lines.

3.2.3 Representations

The rules are, of course, of the same form as used by DENDRAL that was described earlier.

3.2.4 Sketch of Method

META-DENDRAL, like DENDRAL, uses the generation-and-test framework. The process is organized in three stages: Reinterpret the data and summarize evidence (INTSUM); generate plausible candidates for rules (RULEGEN); test and refine the set of plausible rules (RULEMOD).

INTSUM: gives every data point in every spectrum an interpretation as a possible (highly specific) fragmentation. It then summarizes statistically the "weight of evidence" for fragmentations and for atomic configurations that cause these fragmentations. Thus, the job of INTSUM is to translate data to DENDRAL subgraphs and bond-breaks, and to summarize the evidence accordingly.

RULEGEN: conducts a heuristic search of the space of all rules that are legal under the DENDRAL rule syntax and the user-supplied constraints. It searches for plausible rules, i.e. those for which positive evidence exists. A search path is pruned when there is no evidence for rules of the class just generated. The search tree begins with the (single) most general rule (loosely put, "anything" fragments from "anything") and proceeds level-by-level toward more detailed specifications of the "anything." The heuristic stopping criterion measures whether a rule being generated has become too specific, in particular whether it is applicable to too few molecules of the input set. Similarly there is a criterion for deciding whether an emerging rule is too general. Thus, the output of RULEGEN is a set of candidate rules for which there is positive evidence.

RULEMOD: tests the candidate rule set using more complex criteria, including the presence of negative evidence. It removes redundancies in the candidate rule set; merges rules that are supported by the same evidence; tries further specialization of candidates to remove negative evidence; and tries further generalization that preserves positive evidence.

3.2.5 Results

META-DENDRAL produces rule sets that rival in quality those produced by our collaborating experts. In some tests, META-DENDRAL recreated rule sets that we had previously acquired from our experts during the DENDRAL project. In a more stringent test involving members of a family of complex ringed molecules for which the mass spectral theory had not been completely worked out by chemists, META-DENDRAL discovered rule sets for each subfamily. The rules were judged by experts to be excellent and a paper describing them was recently published in a major chemical journal (Buchanan, Smith, et al, 1976).

In a test of the generality of the approach, a version of the META-DENDRAL program is currently being applied to the discovery of rules for the analysis of nuclear magnetic resonance data.

3.3 MYCIN and TEIRESIAS: Medical Diagnosis

3.3.1 Historical note

MYCIN originated in the Ph.D. thesis of E. Shortliffe (now Shortliffe, M.D. as well), in collaboration with the Infectious Disease group at the Stanford Medical School (Shortliffe, 1976). TEIRESIAS, the Ph.D. thesis work of R. Davis, arose from issues and problems indicated by the MYCIN project but generalized by Davis beyond the bounds of medical diagnosis applications (Davis, 1976). Other MYCIN-related theses are in progress.

3.3.2 Tasks

The MYCIN performance task is diagnosis of blood infections and meningitis infections and the recommendation of drug treatment. MYCIN conducts a consultation (in English) with a physician-user about a patient case, constructing lines-of-reasoning leading to the diagnosis and treatment plan.

The TEIRESIAS knowledge acquisition task can be described as follows:

In the context of a particular consultation, confront the expert with a diagnosis with which he does not agree. Lead him systematically back through the line-of-reasoning that produced the diagnosis to the point at which he indicates the analysis went awry. Interact with the expert to modify offending rules or to acquire new rules. Rerun the consultation to test the solution and gain the expert's concurrence.

3.3.3 Representations:

MYCIN's rules are of the form:

IF <conjunctive clauses> THEN <implication>

Here is an example of a MYCIN rule for blood infections.

RULE 85

IF:
1) The site of the culture is blood, and
2) The gram stain of the organism is gramneg, and
3) The morphology of the organism is rod, and
4) The patient is a compromised host

THEN:
There is suggestive evidence (.6) that the identity of the organism is pseudomonas-aeruginosa

----

TEIRESIAS allows the representation of MYCIN-like rules governing the use of other rules, i.e. rule-based strategies. An example follows.

METARULE 2

IF:
1) the patient is a compromised host, and
2) there are rules which mention in their premise pseudomonas
3) there are rules which mention in their premise klebsiellas

THEN:
There is suggestive evidence (.4) that the former should be done before the latter.

3.3.4 Sketch of method

MYCIN employs a generation-and-test procedure of a familiar sort. The generation of steps in the line-of-reasoning is accomplished by backward chaining of the rules. An IF-side clause is either immediately true or false (as determined by patient or test data entered by the physician in the consultation); or is to be decided by subgoaling. Thus, "test" is interleaved with "generation" and serves to prune out incorrect lines-of-reasoning.

Each rule supplied by an expert has associated with it a "degree of certainty" representing the expert's confidence in the validity of the rule (a number from 1 to 10). MYCIN uses a particular ad-hoc but simple model of inexact reasoning to cumulate the degrees of certainty of the rules used in an inference chain (Shortliffe and Buchanan, 1975).

It follows that there may be a number of "somewhat true" lines-of-reasoning -- some indicating one diagnosis, some indicating another. All (above a threshold) are used by the system as sources of knowledge indicating plausible lines-of-reasoning.

TEIRESIAS' rule acquisition process is based on a record of MYCIN's search. Rule acquisition is guided by a set of rule models that dictate the form and indicate the likely content of new rules. Rule models are not given in advance, but are inferred from the knowledge base of existing rules.

3.3.5 Language of Interaction

The language used looks like it might be English but is actually the dialect "Doctor-ese" used by members of the tribe of healing arts practitioners. Reasonably simple language processing methods suffice. When ambiguities of interpretation are encountered, they are fed back to the user for decisions about meaning.

3.3.6 Example diagnosis and treatment plan

The following is an example of a diagnosis reported by MYCIN to one of its users following a consultation. An example consultation will not be shown, but one can be found in Shortliffe's book (Shortliffe, 1976).

My therapy recommendations will be designed to treat for organisms that are either very likely or, although less likely, would have a significant effect on therapy selection if they were present. It is important to

cover for the following probable
infection(s) and associated organism(s):

INFECTION-1 is CYSTITIS
<Item 1> PSEUDOMONAS-CEPACIA [ORGANISM-5]
<Item 2> CITROBACTER-DIVERSUS [ORGANISM-4]
<Item 3> E.COLI [ORGANISM-3]

INFECTION-2 is UPPER-RESPIRATORY-INFECTION
<Item 4> LISTERIA   [ORGANISM-2]

INFECTION-3 is BACTEREMIA
<Item 5> ENTEROCOCCUS [ORGANISM-1]

[REC-1] My preferred therapy recommendation
is as follows:

In order to cover for Items <1 2 3 4 5>:
Give the following in combination:
1) KANAMYCIN
Dose: 750 mg (7.5 mg/kg) q12h IM (or IV)
      for 28 days
Comments: Modify dose in renal failure
2) PENICILLIN
Dose: 2,500,000 units (25000 units/kg)
      q4h IV for 28 days

3.3.7 Results

    MYCIN is currently being prepared for
clinical evaluation. In a preliminary evaluation,
done outside of the clinical setting, a panel of
expert judges was asked to rate MYCIN's
performance along a number of dimensions (e.g.
infecting organism identification, organism
significance). In 90% of the cases submitted to
the judges, a majority of the judges said that the
program's decisions were the-same-as or as-good-as
the decisions they would have made.

3.3.8 Explanation

    MYCIN can expose its line of reasoning by
offering explanations of various kinds. "Why" and
"How" questions can be asked by the user during
the consultation. "Why" is the user's way of
inquiring, "Why do you want to know that?" "How"
is his way of asking, "How did you conclude
that...?" In a post-consultation, he can similarly
inquire about how the diagnostic conclusions were
reached, and inquire about the reasons why certain
other conclusions were ruled out. Examples of the
MYCIN-TEIRESIAS explanation capabilities follow:

MYCIN: Is the patient's illness with
ORGANISM-1 a hospital-acquired infection?
USER: WHY
TEIRESIAS: (i.e. WHY is it important to
determine whether or not the infection with
ORGANISM-1 was acquired while the patient
was hospitalized?)

[1.0] This will aid in determining the
category of ORGANISM-1.

It has already been established that

[1.1] the gram stain of ORGANISM-1 is
gramneg, and
[1.2] the morphology of ORGANISM-1 is rod,
and
[1.3] the aerobicity of ORGANISM-1 is
facultative

Therefore, if

[1.4] the infection with ORGANISM-1 was not
acquired while the patient was hospitalized

then

there is strongly suggestive evidence (.8)
that the category of ORGANISM-1 is
enterobacteriaceae

Also: there is weakly suggestive evidence
(.1) that the identity of
ORGANISM-1 is pseudomonas-aeruginosa
[RULE 37]

USER: HOW 1.3

TEIRESIAS: [i.e. HOW was it established
that the aerobicity of ORGANISM-1 is
facultative?]

The following were used:

[3.1] RULE 27 indicated there is strongly
suggestive evidence (.8) that the
aerobicity of ORGANISM-1 is facultative

[3.2] RULE 27 indicated there is weakly
suggestive evidence (.2) that the
aerobicity of ORGANISM-1 is anaerobic

Since this gave a cumulative CF of (.8) for
facultative, and (.2) for anaerobic, it has
been established that the aerobicity of
ORGANISM-1 is facultative.

USER: WHY DIDN'T YOU CONSIDER
STREPTOCOCCUS AS A POSSIBILITY FOR
ORGANISM-1?

TEIRESIAS: The following rule could have
been used to determine that the identify of
ORGANISM-1 was streptococcus: RULE 33

But clause 2 ("the morphology of the
organism is coccus") was already known
to be false for ORGANISM-1, so the rule was
never tried.

3.3.9 Discussion

A few short comments aimed at knowledge
engineering are in order.

English is the language most natural and
comfortable for our experts and users, and was
therefore the language chosen for interactive
consultation, explanation, and external
representation of the rules (the internal format
is INTERLISP). This situation is not peculiar to
doctors; in most areas of application of
intelligent agents I believe that English (i.e.
natural language) will be the language of choice.
Programming an English language processor and
front-end to such systems is not a scary
enterprise because:

a) the domain is specialized, so that
possible interpretations are constrained.

b) specialist-talk is replete with standard
jargon and stereotyped ways of expressing
knowledge and queries — just right for text
templates, simple grammars and other simple
processing schemes.

c) the ambiguity of interpretation resulting
from simple schemes can be dealt with easily by
feeding back interpretations for confirmation. If
this is done with a pleasant "I didn't quite
understand you..." tone, it is not irritating to
the user.

English may be exactly the wrong language
for representation and interaction in some
domains. It would be awkward, to say the least, to
represent DENDRAL's chemical structures and
knowledge of mass spectrometry in English, or to
interact about these with a user.

Simple explanation schemes have been a part
of the AI scene for a number of years and are not
hard to implement. Really good models of what
explanation is as a transaction between user and
agent, with programs to implement these models,
will be the subject (I predict) of much future
research in AI.

Without the explanation capability, I
assert, user acceptance of MYCIN would have been
nil, and there would have been a greatly
diminished effectiveness and contribution of our
experts.

MYCIN was the first of our programs that
forced us to deal with what we had always
understood: that experts' knowledge is uncertain
and that our inference engines had to be made to
reason with this uncertainty. It is less important
that the inexact reasoning scheme be formal,
rigorous, and uniform than it is for the scheme to
be natural to and easily understandable by the
experts and users.

All of these points can be summarized by
saying that MYCIN and its TEIRESIAS adjunct are
experiments in the design of a see-through system,
whose representations and processes are almost
transparently clear to the domain specialist.
"Almost" here is equivalent to "with a few minutes
of introductory description." The various pieces
of MYCIN — the backward chaining, the English
transactions, the explanations, etc. — are each
simple in concept and realization. But there are
great virtues to simplicity in system design; and
viewed as a total intelligent agent system,
MYCIN/TEIRESIAS is one of the best engineered.

3.4 SU/X: signal understanding

3.4.1 Historical note

SU/X is a system design that was tested in
an application whose details are classified.
Because of this, the ensuing discussion will
appear considerably less concrete and tangible
than the preceding case studies. This system
design was done by H.P. Nii and me, and was
strongly influenced by the CMU Hearsay II system
design.

3.4.2 Task

SU/X's task is the formation and continual
updating, over long periods of time, of hypotheses
about the identity, location, and velocity of
objects in a physical space. The output desired is
a display of the "current best hypotheses" with
full explanation of the support for each. There
are two types of input data: the primary signal
(to be understood); and auxiliary symbolic data
(to supply context for the understanding). The
primary signals are spectra, represented as
descriptions of the spectral lines. The various
spectra cover the physical space with some spatial
overlap.

3.4.3 Representations

The rules given by the expert about objects, their behavior, and the interpretation of signal data from them are all represented in the situation => action form. The "situations" constitute invoking conditions and the "actions" are processes that modify the current hypotheses, post unresolved issues, recompute evaluations, etc. The expert's knowledge of how to do analysis in the task is also represented in rule form. These strategy rules replace the normal executive program.

The situation-hypothesis is represented as a node-link graph, tree-like in that it has distinct "levels," each representing a degree of abstraction (or aggregation) that is natural to the expert in his understanding of the domain. A node represents an hypothesis; a link to that node represents support for that hypothesis (as in HEARSAY II, "support from above" or "support from below"). "Lower" levels are concerned with the specifics of the signal data. "Higher" levels represent symbolic abstractions.

3.4.4 Sketch of method

The situation-hypothesis is formed incrementally. As the situation unfolds over time, the triggering of rules modifies or discards existing hypotheses, adds new ones, or changes support values. The situation-hypothesis is a common workspace ("blackboard," in HEARSAY jargon) for all the rules.

In general, the incremental steps toward a more complete and refined situation-hypothesis can be viewed as a sequence of local generate-and-test activities. Some of the rules are plausible move generators, generating either nodes or links. Other rules are evaluators, testing and modifying node descriptions.

In typical operation, new data is submitted for processing (say, N time-units of new data). This initiates a flurry of rule-triggerings and consequently rule-actions (called "events"). Some events are direct consequences of the data; other events arise in a cascade-like fashion from the triggering of rules. Auxiliary symbolic data also cause events, usually affecting the higher levels of the hypothesis. As a consequence, support-from-above for the lower level processes is made available; and expectations of possible lower level events can be formed. Eventually all the relevant rules have their say and the system becomes quiescent, thereby triggering the input of new data to re-energize the inference activity.

The system uses the simplifying strategy of maintaining only one "best" situation-hypothesis at any moment, modifying it incrementally as required by the changing data. This approach is made feasible by several characteristics of the

domain. First, there is the strong continuity over time of objects and their behaviors (specifically, they do not change radically over time, or behave radically differently over short periods). Second, a single problem (identity, location and velocity of a particular set of objects) persists over numerous data gathering periods. (Compare this to speech understanding in which each sentence is spoken just once, and each presents a new and different problem.) Finally, the system's hypothesis is typically "almost right," in part because it gets numerous opportunities to refine the solution (i.e. the numerous data gathering periods), and in part because the availability of many knowledge sources tends to over-determine the solution. As a result of all of these, the current best hypothesis changes only slowly with time, and hence keeping only the current best is a feasible approach.

Of interest are the time-based events. These rule-like expressions, created by certain rules, trigger upon the passage of specified amounts of time. They implement various "wait-and-see" strategies of analysis that are useful in the domain.

3.4.5 Results

In the test application, using signal data generated by a simulation program because real data was not available, the program achieved expert levels of performance over a span of test problems. Some problems were difficult because there was very little primary signal to support inference. Others were difficult because too much signal induced a plethora of alternatives with much ambiguity.

A modified SU/X design is currently being used as the basis for an application to the interpretation of x-ray crystallographic data, the CRYSALIS program mentioned later.

3.4.6 Discussion

The role of the auxiliary symbolic sources of data is of critical importance. They supply a symbolic model of the existing situation that is used to generate expectations of events to be observed in the data stream. This allows flow of inferences from higher levels of abstraction to lower. Such a process, so familiar to AI researchers, apparently is almost unrecognized among signal processing engineers. In the application task, the expectation-driven analysis is essential in controlling the combinatorial processing explosion at the lower levels, exactly the explosion that forces the traditional signal processing engineers to seek out the largest possible number-cruncher for their work.

The design of appropriate explanations for the user takes an interesting twist in SU/X. The

situation-hypothesis unfolds piecemeal over time, but the "appropriate" explanation for the user is one that focuses on individual objects over time. Thus the appropriate explanation must be synthesized from a history of all the events that led up to the current hypothesis. Contrast this with the MYCIN-TEIRESIAS reporting of rule invocations in the construction of a reasoning chain.

Since its knowledge base and its auxiliary symbolic data give it a model-of-the-situation that strongly constrains interpretation of the primary data stream, SU/X is relatively unperturbed by errorful or missing data. These data conditions merely cause fluctuations in the credibility of individual hypotheses and/or the creation of the "wait-and-see" events. SU/X can be (but has not yet been) used to control sensors. Since its rules specify what types and values of evidence are necessary to establish support, and since it is constantly processing a complete hypothesis structure, it can request "critical readings" from the sensors. In general, this allows an efficient use of limited sensor bandwidth and data acquisition processing capability.

3.5 OTHER CASE STUDIES

Space does not allow more than just a brief sketch of other interesting projects that have been completed or are in progress.

3.5.1 AM: mathematical discovery

AM is a knowledge-based system that conjectures interesting concepts in elementary mathematics. It is a discoverer of interesting theorems to prove, not a theorem proving program. It was conceived and executed by D. Lenat for his Ph.D. thesis, and is reported by him in these proceedings ("An Overview of AM").

AM's knowledge is basically of two types: rules that suggest possibly interesting new concepts from previously conjectured concepts; and rules that evaluate the mathematical "interestingness" of a conjecture. These rules attempt to capture the expertise of the professional mathematician at the task of mathematical discovery. Though Lenat is not a professional mathematician, he was able successfully to serve as his own expert in the building of this program.

AM conducts a heuristic search through the space of concepts creatable from its rules. Its basic framework is generation-and-test. The generation is plausible move generation, as indicated by the rules for formation of new concepts. The test is the evaluation of "interestingness." Of particular note is the method of test-by-example that lends the flavor of

scientific hypothesis testing to the enterprise of mathematical discovery.

Initialized with concepts of elementary set theory, it conjectured concepts in elementary number theory, such as "add," "multiply" (by four distinct paths!), "primes," the unique factorization theorem, and a concept similar to primes but previously not much studied called "maximally divisible numbers."

3.5.2 MOLGEN: planning experiments in molecular genetics

MOLGEN a collaboration with the Stanford Genetics Department, is work in progress. MOLGEN's task is to provide intelligent advice to a molecular geneticist on the planning of experiments involving the manipulation of DNA. The geneticist has various kinds of laboratory techniques available for changing DNA material (cuts, joins, insertions, deletions, and so on); techniques for determining the biological consequences of the changes; various instruments for measuring effects; various chemical methods for inducing, facilitating, or inhibiting changes; and many other tools.

MOLGEN will offer planning assistance in organizing and sequencing such tools to accomplish an experimental goal. In addition MOLGEN will check user-provided experiment plans for feasibility; and its knowledge base will be a repository for the rapidly expanding knowledge of this specialty, available by interrogation.

Current efforts to engineer a knowledge-base management system for MOLGEN are described by Martin et al in a paper in these proceedings. This subsystem uses and extends the techniques of the TEIRESIAS system discussed earlier.

In MOLGEN the problem of integration of many diverse sources of knowledge is central since the essence of the experiment planning process is the successful merging of biological, genetic, chemical, topological, and instrument knowledge. In MOLGEN the problem of representing processes is also brought into focus since the expert's knowledge of experimental strategies -- proto-plans -- must also be represented and put to use.

3.5.3 CRYSALIS: inferring protein structure from electron density maps

CRYSALIS, too, is work in progress. Its task is to hypothesize the structure of a protein from a map of electron density that is derived from x-ray crystallographic data. The map is three-dimensional, and the contour information is crude and highly ambiguous. Interpretation is guided and supported by auxiliary information, of which the amino acid sequence of the protein's backbone is the most important. Density map interpretation

is a protein chemist's art. As always, capturing this art in heuristic rules and putting it to use with an inference engine is the project's goal.

The inference engine for CRYSALIS is a modification of the SU/X system design described above. The hypothesis formation process must deal with many levels of possibly useful aggregation and abstraction. For example, the map itself can be viewed as consisting of "peaks," or "peaks and valleys," or "skeleton." The protein model has "atoms," "amide planes," "amino acid sidechains," and even massive substructures such as "helices." Protein molecules are so complex that a systematic generation-and-test strategy like DENDRAL's is not feasible. Incremental piecing together of the hypothesis using region-growing methods is necessary.

The CRYSALIS design (alias SU/P) is described in a recent paper by Nii and Feigenbaum (1977).

4 SUMMARY OF CASE STUDIES

Some of the themes presented earlier need no recapitulation, but I wish to revisit three here: generation-and-test; situation => action rules; and explanations.

4.1 Generation and Test

Aircraft come in a wide variety of sizes, shapes, and functional designs and they are applied in very many ways. But almost all that fly do so because of the unifying physical principle of lift by airflow; the others are described by exception. So it is with intelligent agent programs and, the information processing psychologists tell us, with people. One unifying principle of "intelligence" is generation-and-test. No wonder that it has been so thoroughly studied in AI research!

In the case studies, generation is manifested in a variety of forms and processing schemes. There are legal move generators defined formally by a generating algorithm (DENDRAL's graph generating algorithm); or by a logical rule of inference (MYCIN's backward chaining). When legal move generation is not possible or not efficient, there are plausible move generators (as in SU/X and AM). Sometimes generation is interleaved with testing (as in MYCIN, SU/X, and AM). In one case, all generation precedes testing (DENDRAL). One case (META-DENDRAL) is mixed, with some testing taking place during generation, some after.

Test also shows great variety. There are simple tests (MYCIN: "Is the organism aerobic?"; SU/X: "Has a spectral line appeared at position P?") Some tests are complex heuristic evaluations (AM: "Is the new concept 'interesting'?"; MOLGEN:

"Will the reaction actually take place?") Sometimes a complex test can involve feedback to modify the object being tested (as in META-DENDRAL).

The evidence from our case studies supports the assertion by Newell and Simon that generation-and-test is a law of our science (Newell and Simon, 1976).

4.2 Situation - > Action rules

Situation => Action rules are used to represent experts' knowledge in all of the case studies. Always the situation part indicates the specific conditions under which the rule is relevant. The action part can be simple (MYCIN: conclude presence of particular organism; DENDRAL: conclude break of particular bond). Or it can be quite complex (MOLGEN: an experiential procedure). The overriding consideration in making design choices is that the rule form chosen be able to represent clearly and directly what the expert wishes to express about the domain. As illustrated, this may necessitate a wide variation in rule syntax and semantics.

From a study of all the projects, a regularity emerges. A salient feature of the Situation => Action rule technique for representing expert's knowledge is the modularity of the knowledge base, with the concomitant flexibility to add or change the knowledge easily as the experts' understanding of the domain changes. Here too one must be pragmatic, not doctrinaire. A technique such as this can not represent modularity of knowledge if that modularity does not exist in the domain. The virtue of this technique is that it serves as a framework for discovering what modularity exists in the domain. Discovery may feed back to cause reformulation of the knowledge toward greater modularity.

Finally, our case studies have shown that strategy knowledge can be captured in rule form. In TEIRESIAS, the metarules capture knowledge of how to deploy domain knowledge; in SU/X, the strategy rules represent the experts' knowledge of "how to analyze" in the domain.

4.3 Explanation

Most of the programs, and all of the more recent ones, make available an explanation capability for the user, be he end-user or system developer. Our focus on end-users in applications domains has forced attention to human engineering issues, in particular making the need for the explanation capability imperative.

The Intelligent Agent viewpoint seems to us to demand that the agent be able to explain its activity; else the question arises of who is in

control of the agent's activity. The issue is not academic or philosophical. It is an engineering issue that has arisen in medical and military applications of intelligent agents, and will govern future acceptance of AI work in applications areas. And on the philosophical level one might even argue that there is a moral imperative to provide accurate explanations to end-users whose intuitions about our systems are almost nil.

Finally, the explanation capability is needed as part of the concerted attack on the knowledge acquisition problem. Explanation of the reasoning process is central to the interactive transfer of expertise to the knowledge base, and it is our most powerful tool for the debugging of the knowledge base.

5 EPILOGUE

What we have learned about knowledge engineering goes beyond what is discernible in the behavior of our case study programs. In the next paper of this two-part series, I will raise and discuss many of the general concerns of knowledge engineers, including these:

What constitutes an "application" of AI techniques?

There is a difference between a serious application and an application-flavored toy problem.

What are some criteria for the judicious selection of an application of AI techniques?

What are some applications areas worthy of serious attention by knowledge engineers?

For example, applications to science, to signal interpretation, and to human interaction with complex systems.

How to find and fascinate an Expert.

The background and prior training of the expert.

The level of commitment that can be elicited.

Designing systems that "think the way I do."

Sustaining attention by quick feedback and incremental progress.

Focusing attention to data and specific problems.

Providing ways to express uncertainty of expert knowledge.

The side benefits to the expert of his investment in the knowledge engineering activity.

Gaining consensus among experts about the knowledge of a domain.

The consensus may be a more valuable outcome of the knowledge engineering effort than the building of the program.

Problems faced by knowledge engineers today:

The lack of adequate and appropriate computer hardware.

The difficulty of export of systems to end-users, caused by the lack of properly-sized and -packaged combinations of hardware and software

The chronic absence of cumulation of AI techniques in the form of software packages that can achieve wide use.

The shortage of trained knowledge engineers.

The difficulty of obtaining and sustaining funding for interesting knowledge engineering projects.

6 ACKNOWLEDGMENT

7 REFERENCES

General

Feigenbaum, E.A. "Artificial Intelligence Research: What is it? What has it achieved? Where is it going?," invited paper, Symposium on Artificial Intelligence, Canberra, Australia, 1974.

Goldstein, I. and S. Papert, "Artificial Intelligence, Language, and the Study of Knowledge," Cognitive Science, Vol.1, No.1, 1977.

Gregory, R., "On How so Little Information Controls so Much Behavior," Bionics Research Report No. 1, Machine Intelligence Department, University of Edinburgh, 1968.

Newell, A. and H.A. Simon, Human Problem Solving, Prentice-Hall, 1972.

Newell, A. and H.A. Simon, "Computer Science as Empirical Inquiry: Symbols and Search," Com ACM, 19, 3, March, 1976.

DENDRAL and META-DENDRAL

Feigenbaum, E.A., Buchanan, B.G. and J. Lederberg, "On Generality and Problem Solving: a Case Study Using the DENDRAL Program," Machine Intelligence 6, Edinburgh Univ. Press, 1971.

Buchanan, B.G., Duffield, A.M. and A.V. Robertson, "An Application of Artificial Intelligence to the Interpretation of Mass Spectra," Mass Spectrometry Techniques and Applications, G.W.A. Milne, Ed., John Wiley & Sons, Inc., p. 121, 1971.

Michie, D. and B.G. Buchanan, "Current Status of the Heuristic DENDRAL Program for Applying Artificial Intelligence to the Interpretation of Mass Spectra," Computers for Spectroscopy, R.A.G. Carrington, ed., London: Adam Hilger, 1974.

Buchanan, B.G., "Scientific Theory Formation by Computer," Nato Advanced Study Institutes Series, Series E: Applied Science, 14:515, Noordhoff-Leyden, 1976.

Buchanan, B.G., Smith, D.H., White, W.C., Gritter, R.J., Feigenbaum, E.A., Lederberg, J. and C. Djerassi, "Applications of Artificial Intelligence for Chemical Inference XXII. Automatic Rule Formation in Mass Spectrometry by Means of the Meta-DENDRAL Program," Journal of the ACS, 98:6168, 1976.

MYCIN

Shortliffe, E. Computer-based Medical Consultations: MYCIN, New York, Elsevier, 1976.

Davis, R., Buchanan, B.G. and E.H. Shortliffe, "Production Rules as a Representation for a Knowledge-Based Consultation Program," Artificial Intelligence, 8, 1, February, 1977.

Shortliffe, E.H. and B.G. Buchanan, "A Model of Inexact Reasoning in Medicine," Mathematical Biosciences, 23:351, 1975.

TEIRESIAS

Davis, R., "Applications of Meta Level Knowledge to the Construction, Maintenance and Use of Large Knowledge Bases," Memo HPP-76-7, Stanford Computer Science Department, Stanford, CA, 1976.

Davis, R., "Interactive Transfer of Expertise I: Acquisition of New Inference Rules," these Proceedings.

Davis, R. and B.G. Buchanan, "Meta-Level Knowledge: Overview and Applications," these Proceedings.

SU/X

Nii, H.P. and E.A. Feigenbaum, "Rule Based Understanding of Signals," Proceedings of the Conference on Pattern-Directed Inference Systems, 1977 (forthcoming), also Memo HPP-77-7, Stanford Computer Science Department, Stanford, CA, 1977.

AM

Lenat, D., "AM: An Artificial Intelligence Approach to Discovery in Mathematics as Heuristic Search," Memo HPP-76-8, Stanford Computer Science Department, Stanford, CA, 1976.

MOLGEN

Martin, N., Friedland, P., King, J., and M. Stefik, "Knowledge Base Management for Experiment Planning in Molecular Genetics," these Proceedings.

CRYSALIS

Engelmore, R. and H.P. Nii, "A Knowledge-Based System for the Interpretation of Protein X-Ray Crystallographic Data," Memo HPP-77-2, Department of Computer Science, Stanford, CA, 1977.

References

1. Adams, J.B. A probability model of medical reasoning and the MYCIN model. Math. Biosci. 32,177-186 (1976).

2. Anderson, R.H., Gallegos, M., Gillogly, J.J., Greenberg, R., and Villanueva, R. RITA Reference Manual, Report R-1808-ARPA, The Rand Corporation, Santa Monica, CA., September 1977.

3. Bennett J.S., Creary L.G., Engelmore R.E., Melosh R.B., A Knowledge-based Consultant for structural analysis, forthcoming.

4. Bleich, H.L. The computer as a consultant. New Eng. J. Med. 284,141-147 (1971).

5. Blum, Robert L. and Wiederhold, Gio: Inferring Knowledge from Clinical Data Banks Utilizing Techniques from Artificial Intelligence. "Proc. 2nd Annual Symp. on Comp. Applic. in Med. Care," pp. 303-307, IEEE, Washington D.C., Nov. 5-9, 1978.

6. Bobrow D.G., Winograd T., An Overview of KRL, a Knowledge Representation Language, Cognitive Science 1:1 (1977).

7. Bobrow D.G., Winograd T., Experience with KRL-0, One cycle of a knowledge representation language, Proceedings of the 5th International Joint Conference on Artificial Intelligence, Cambridge, Mass. (August 1977).

8. Bonnet A., BAOBAB, A parser for a rule-based system using a semantic grammar, Technical Report HPP-78-10, Heuristic Programming Project, Stanford California (September 1978).

9. Brown, J.S., Steps toward a Theoretic Foundation for Complex, Knowledge-Based CAI. BBN No. 3135.

10. Brown, J.S., Collins, A., and Harris, G.

Artificial Intelligence and Learning Strategies. To appear
in _Learning Strategies_ (ed. Harry O'Neil), Academic
Press, New York, 1978.

11. Buchanan, Bruce G. and Feigenbaum, Fdward A.   DENDRAL and
     Meta-DENDRAL: Their Applications Dimension, Artificial
     Intelligence, 11:5 (1978).

12. Clancey, W. "The Structure of a Case Method Dialogue", to
     appear in Int. Jnl. of Man Machine Studies, Fall, 1978.

13.   Colby, K.M., Weber, S., and Hilf, F. Artificial
      paranoia. Artificial Intelligence 2,1-25 (1971).

14.   Croft, D.J.   Is computerized diagnosis possible? Comput.
      Biomed. Res. 5,351-367 (1972).

15.   Davis, R. Applications Of Meta Level Knowledge To
      the Construction, Maintenance, And Use Of Large Knowledge
      Bases. Doctoral dissertation, Stanford University ; Memo
      HPP-76-7, Stanford Computer Science Department, 1976.

16.   Davis, R. and King, J. An overview of production
      systems. Machine Intelligence 8: Machine Representations of
      Knowledge (eds. E.W. Elcock and D. Michie), John Wiley,
      April 1977.

17.   de Dombal, F.T., Leaper, D.J., Staniland, J.R., McCann,
      A.P., Horrocks, J.C. Computer aided diagnosis of acute
      abdominal pain. Brit. Med. J. II,9-13 (1972).

18. Duda, R. O., Hart, P., Nilsson, N. & Sutherland, G.
     "Semantic network representations in rule-based inference
     systems", in _Pattern Directed Inference Systems_ (eds.
     Waterman and Hayes-Roth), Academic Press,New York, 1978.

19. Engelmore R.S., Nii H.P., A knowledge-based system for the
     interpretation of protein x-ray crystallographic data,
     Heuristic Programming Project Memo HPP-77-2 (February
     1977).

20. Erman L.D., Lesser V.R., A multi-level organization for
     problem solving using many, diverse, cooperating sources of

knowledge, in Proceedings of the 4th International Joint Conference on Artificial Intelligence, Tbilsi, Russia (1975).

21. Fagan L.M., Ventilator Manager: A program to provide on-line consultative advice in the intensive care unit, Heuristic Programming Project Memo HPP-78-16 (Working Paper), Computer Science Department, Stanford University (September 1978).

22. Feigenbaum E.A., The art of artificial intelligence: I. Themes and case studies of knowledge engineering, Proceedings of the 5th International Joint Conference on Artificial Intelligence, Cambridge, Mass. (August 1977).

23. Feitelson J., Stefik M., A case study of the reasoning in a genetics experiment, Heuristic Programming Project Report 77-18 (working paper) ,Computer Science Department, Stanford University (April 1977).

24. Friedman, R.B. and Gustafson, D.H. Computers in clinical medicine: a critical review. Comput. Biomed. Res. 10,199-204 (1977).

25. Fries, J. Time-oriented patient records and a computer data-bank. J. Amer. Med. Assoc. 222,1536-1542 (1973).

26. Goldstein, I., Papert, S. Artificial Intelligence, Language, and study of knowledge. Cognitive Science 1:1 (1977).

27. Gorry, G.A. and Barnett, G.O. Experience with a model of sequential diagnosis. Comput. Biomed. Res. 1,490-507 (1968).

28. Gorry, G.A., Kassirer, J.P., Essig, A., and Schwartz, W.B. Decision analysis as the basis for computer-aided management of acute renal failure. Amer. J. Med. 55,473-484 (1973).

29. Gorry, G.A., Silverman, H., and Pauker, S.G. Capturing clinical expertise: a computer program that considers clinical responses to digitalis. Amer. J. Med. 64,452-460 (1978).

30. Green, B.F., Wolf, A.K., Chomsky, C., and Laughery, K. BASEBALL: An automatic question-answerer. In Computers and Thought (eds. E.A. Feigenbaum and J. Feldman), pp. 207-216, McGraw-Hill, San Francisco,1963.

31. Harless, W.G., Drennon, G.G., Marxer, J.J., Root, J.A., Wilson, L.L., and Miller, G.E. CASE - a natural language computer model. Comput. Biol. Med. 3,227-246 (1973).

32. Hart, P.E. Progress on a computer-based consultant. AI Technical Note 99, Stanford Research Institute, Menlo Park, CA., January 1975.

33. Hayes-Roth F., Lesser V.R., Focus of attention in the HEARSAY-II speech understanding system, Proceedings of the 5th International Joint Conference on Artificial Intelligence, Cambridge, Mass. (August 1977).

34. Heiser J.F., Brooks R.E., Ballard J.P., "Progress Report: A Computerized Psychopharmacology Advisor", Proceedings of the 11th Colegium Internationale NeuroPsychopharmacologicum. Vienna, 1978.

35. Heiser, J.F. and Brooks, R.E. A computerized psychopharmacology advisor. Proceedings of the 4th Annual AIM Workshop, Rutgers University, June 1978.

36. Hoffer, E.P. Experience with the use of computer simulation models in medical education. Comput. Biol. Med. 3,269-279 (1973).

37. Kunz J.C., Fallat R.J., McClung D.H., Osborn J.J., Votteri B.A., Nii H.P., Aikins J.S., Fagan L.M., Feigenbaum E.A., A physiological rule based system for interpreting pulmonary function test results, Heuristic Programming Project Memo HPP-78-19, Stanford University, 1978.

38. Lenat D.B., The ubiquity of discovery, Artificial Intelligence 9:3 (1977).

39. Lowerre B.T., The HARPY speech recognition system, Doctoral thesis, Department of Computer Science, Carnegie-Mellon University (April 1976).

40. Martin N., Friedland P., King J., Stefik M., Knowledge Base Management for Experiment Planning, Proceedings of the 5th International Joint Conference on Artificial Intelligence, Cambridge, Mass. (August 1977).

41. Mesel, E., Wirtshcafter, D.D., Carpenter, J.T., Durant, J.R., Henke, C., and Gray, E.A. Clincial Algorithms for Cancer Chemotherapy — Systems for Community-Based Consultant-Extenders and Oncology Centers. Meth. Inform. Med. 15:3, 168-73 (1976).

42. Minsky M., A framework for representing knowledge, in The psychology of computer vision, (ed. P. Winston), McGraw-Hill, New York (1975).

43. Nii H.P., Feigenbaum E.A., Rule-based understanding of signals in Pattern-Directed Inference Systems (eds. Waterman and Hayes-Roth), Academic Press, New York, 1978.

44. Osborn, J.J., Kunz, J.C., and Fagan, L.M. PUFF/VM: interpretation of physiological measurements in the pulmonary function laboratory and the intensive care unit. Proceedings of the 4th Annual AIM Workshop, Rutgers University, June 1978.

45. Pauker, S.G., Gorry, G.A., Kassirer, J.P., and Schwartz, W.B. Towards the simulation of clinical cognition: taking a present illness by computer. Amer. J. Med. 60,981-996 (1976).

46. Pople, H.E., Myers, J.D., Miller, R.A. DIALOG (INTERNIST): a model of diagnostic logic for internal medicine. Proceedings of the 4th International Joint Conference on Artificial Intelligence, pp. 849-855, Tbilisi, Russia, 1975.

47. Quillian, M.R. Semantic memory. In Semantic Information Processing (ed. M. Minsky), pp. 227-270, M.I.T. Press, Cambridge, MA., 1968.

48. Scott, A.C., Clancey, W.J., Davis, R., and Shortliffe, E.H. Explanation capabilities of knowledge-based production systems. Amer. J. Computational Linguistics, Microfiche 62, 1977.

49.   Shortliffe, E.H. and Buchanan, B.G.   A model of inexact reasoning in medicine. Math. Biosci. 23,351-379 (1975).

50.   Shortliffe, E.H., Davis, R., Axline, S.G., Buchanan, B.G., Green, C.C., and Cohen, S.N.   Computer-based consultations in clinical therapeutics: explanation and rule-acquisition capabilities of the MYCIN system.   Comput. Biomed. Res. 8,303-320 (1975).

51.   Shortliffe, E.H.   Computer-Based Medical Consultations: MYCIN. Elsevier/North Holland, New York, 1976.

52.   Stefik M., An examination of a frame-structured representation system, Stanford Heuristic Programming Project Memo HPP-78-13 (working paper) (September 1978).

53.   Stefik M., Inferring DNA structures from segmentation data, Artificial Intelligence 11 (1978).

54.   Van Melle, W.   Would you like advice on another horn? MYCIN project internal working paper, Stanford University, Stanford, California, December 1974.

55.   Warner, H.R., Toronto, A.F., and Veasy, L.G. Experience with Bayes' theorem for computer diagnosis of congenital heart disease. Anns. N.Y. Acad. Sci. 115,558-567 (1964).

56.   Weinberg, A.D. CAI at the Ohio State University College of Medicine. Comput. Biol. Med. 3,299-305 (1973).

57.   Weiss, S., Kulikowski, C.A., and Safir, A. Glaucoma consultation by computer.   Comput. Biol. Med. 8,25-40 (1978).

58.   Weyl, S., Fries, J., Wiederhold, G., and Germano, F. A modular self-describing clinical databank system. Comput. Biomed. Res. 8,279-293 (1975).

59.   Woods, W.A. et al. The lunar sciences natural language information system: final report, BBN Report 2378, Bolt, Beranek and Newman, Cambridge, MA., June 1972.

60. Wooster, H. and Lewis, J.F. Distribution of computer-assisted instruction materials in biomedicine through the Lister Hill Center Experimental Network. Comput. Biol. Med. 3,319-323 (1973).

61. Wortman, P.M. Medical diagnosis: an information processing approach. Comput. Biomed. Res. 5,315-328 (1972).

62. Yu, V.L., Buchanan, B.G., Shortliffe, E.H., Wraith, S.M., Davis, R., Scott, A.C., and Cohen, S.N. Evaluating the performance of a computer-based consultant. To appear in Computer Programs in Biomedicine, 1978.

63. Yu, V.L., Fagan, L.M., Wraith, S.M., Clancey, W.J., Scott, A.C., Hannigan, J., Blum, R.L., Buchanan, B.G., and Cohen, S.N. Computer-based consultation in antimicrobial selection - a comparative evaluation by experts. Submitted for publication, September 1978.

The appropriate programmatic and administrative personnel of each institution involved in this grant application are aware of the NIH consortium grant policy and are prepared to establish the necessary inter-institutional agreement(s) consistent with that policy.